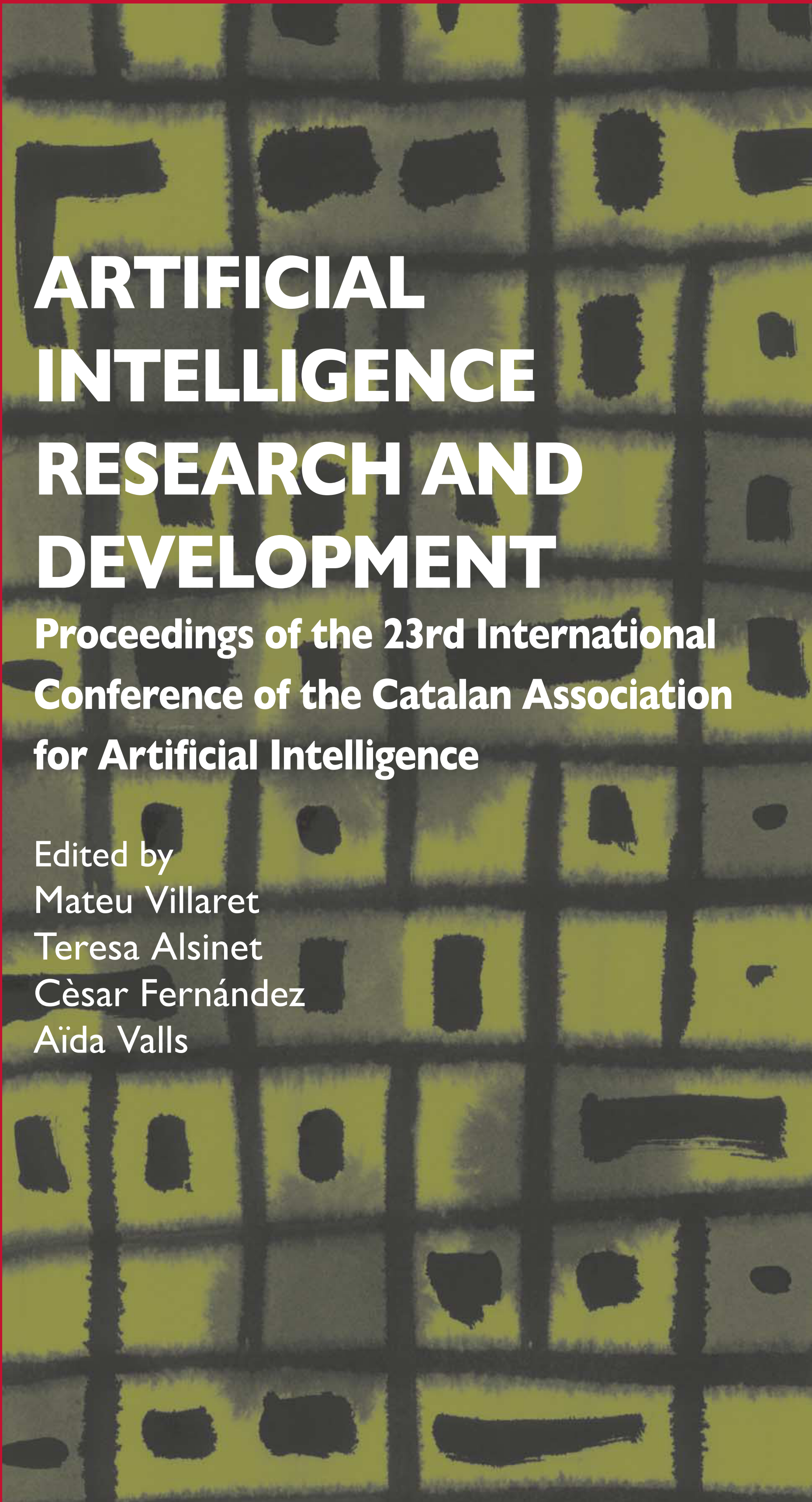
Frontiers
in
Artificial
Intelligence
and
Applications



ARTIFICIAL INTELLIGENCE RESEARCH AND DEVELOPMENT

Proceedings of the 23rd International Conference of the Catalan Association for Artificial Intelligence

Edited by
Mateu Villaret
Teresa Alsinet
Cèsar Fernández
Aïda Valls

IOS Press

ARTIFICIAL INTELLIGENCE RESEARCH AND DEVELOPMENT

Artificial intelligence has become an indispensible part of our lives in recent years, affecting all aspects from business and leisure to transport and health care.

This book presents the proceedings of the 23rd edition of the International Conference of the Catalan Association for Artificial Intelligence (CCIA), an annual event that serves as a meeting point for researchers in Artificial Intelligence in the area of the Catalan speaking territories and from around the world. The 2021 edition was held online as a virtual conference from 20 - 22 October 2021 due to the COVID-19 pandemic. The book contains 42 long papers and 9 short papers, carefully reviewed and selected. The papers cover all aspects of artificial intelligence and are divided under six section headings: combinatorial problem solving and logics for artificial intelligence; sentiment analysis and text analysis; data science and decision support systems; machine learning; computer vision; and explainability and argumentation. Abstracts of the 2 invited talks delivered at the conference by Prof. Patty Kostkova and Prof. João Marques-Silva are also included.

Offering a state of the art overview of the subject from a regional perspective, the book will be of interest to all those working in the field of artificial intelligence.

9 781643 682105

ISBN 978-1-64368-210-5 (print)
ISBN 978-1-64368-211-2 (online)
ISSN 0922-6389 (print)
ISSN 1879-8314 (online)

ARTIFICIAL INTELLIGENCE RESEARCH AND DEVELOPMENT

Frontiers in Artificial Intelligence and Applications

The book series Frontiers in Artificial Intelligence and Applications (FAIA) covers all aspects of theoretical and applied Artificial Intelligence research in the form of monographs, selected doctoral dissertations, handbooks and proceedings volumes. The FAIA series contains several sub-series, including 'Information Modelling and Knowledge Bases' and 'Knowledge-Based Intelligent Engineering Systems'. It also includes the biennial European Conference on Artificial Intelligence (ECAI) proceedings volumes, and other EurAI (European Association for Artificial Intelligence, formerly ECCAI) sponsored publications. The series has become a highly visible platform for the publication and dissemination of original research in this field. Volumes are selected for inclusion by an international editorial board of well-known scholars in the field of AI. All contributions to the volumes in the series have been peer reviewed.
The FAIA series is indexed in ACM Digital Library; DBLP; EI Compendex; Google Scholar; Scopus; Web of Science: Conference Proceedings Citation Index – Science (CPCI-S) and Book Citation Index – Science (BKCI-S); Zentralblatt MATH.

Volume 339

Recently published in this series

Vol. 338. C. Frasson, K. Kabassi and A. Voulodimos (Eds.), Novelties in Intelligent Digital Systems – Proceedings of the 1st International Conference (NIDS 2021), Athens, Greece, September 30 – October 1, 2021
Vol. 337. H. Fujita and H. Perez-Meana (Eds.), New Trends in Intelligent Software Methodologies, Tools and Techniques – Proceedings of the 20th International Conference on New Trends in Intelligent Software Methodologies, Tools and Techniques (SoMeT_21)
Vol. 336. A. Biere, M. Heule, H. van Maaren and T. Walsh (Eds.), Handbook of Satisfiability – Second Edition
Vol. 335. M. Nørskov, J. Seibt and O.S. Quick (Eds.), Culturally Sustainable Social Robotics – Proceedings of Robophilosophy 2020 – August 18–21, 2020, Aarhus University and online
Vol. 334. S. Villata, J. Harašta and P. Křemen (Eds.), Legal Knowledge and Information Systems – JURIX 2020: The Thirty-third Annual Conference, Brno, Czech Republic, December 9–11, 2020
Vol. 333. M. Tropmann-Frick, B. Thalheim, H. Jaakkola, Y. Kiyokir and N. Yoshida (Eds.), Information Modelling and Knowledge Bases XXXII

ISSN 0922-6389 (print)
ISSN 1879-8314 (online)

Artificial Intelligence Research and Development

Proceedings of the 23rd International Conference of the Catalan Association for Artificial Intelligence

Edited by

Mateu Villaret
Universitat de Girona, Spain

Teresa Alsinet
Universitat de Lleida, Spain

Cèsar Fernández
Universitat de Lleida, Spain

and

Aïda Valls
Universitat Rovira i Virgili, Spain

IOS Press

Amsterdam Berlin Washington, DC

ISBN 978-1-64368-210-5 (print)
ISBN 978-1-64368-211-2 (online)
Library of Congress Control Number: 2021946845
doi: 10.3233/FAIA339

Publisher
IOS Press BV
Nieuwe Hemweg 6B
1013 BG Amsterdam
Netherlands
fax: +31 20 687 0019
e-mail: order@iospress.nl

For book sales in the USA and Canada:
IOS Press, Inc.
6751 Tepper Drive
Clifton, VA 20124
USA
Tel.: +1 703 830 6300
Fax: +1 703 830 2300
sales@iospress.com

LEGAL NOTICE

The publisher is not responsible for the use which might be made of the following information.

PRINTED IN THE NETHERLANDS

Preface

The International Conference of the Catalan Association for Artificial Intelligence (CCIA) is an annual event that serves as a meeting point for researchers in Artificial Intelligence based in the area of the Catalan speaking territories (south France, Catalonia, Valencia, Balearic Islands and Alghero in Italy) and from around the world.

This book constitutes the proceedings of the 23rd edition of the CCIA, held in Lleida, in October 2021. The conference was held virtually due to the COVID-19 pandemic. Previous editions of the CCIA have been in Tarragona (1998), Girona (1999), Vilanova i la Geltrú (2000), Barcelona (2001, 2004, 2014, 2016), Castelló de la Plana (2002), Mallorca (2003), L'Alguer (Italy) (2005), Perpinyà (France) (2006), Andorra (2007), Sant Martí d'Empúries (2008), Cardona (2009), L'Espluga de Francolí (2010), Lleida (2011), Alacant (2012), Vic (2013), València (2015), Deltebre (2017), Roses (2018) and Colònia de Sant Jordi (2019). There was no CCIA in 2020 because of the severe restrictions caused by the COVID-19 pandemic.

The 42 long papers and 9 short papers presented in this volume were carefully reviewed and selected from 59 submissions. The reviewing process was made possible thanks to the 93 artificial intelligence experts who make up the program committee, plus some additional referees. We would especially like to thank them for their efforts in this task, as well as thanking the authors of the 59 submissions for their work.

The accepted papers deal with all aspects of artificial intelligence including: Combinatorial Problem Solving and Logics for Artificial Intelligence, Sentiment Analysis and Text Analysis, Data Science and Decision Support Systems, Machine Learning, Computer Vision, and Explainability and Argumentation. This book of proceedings also includes abstracts of the two invited talks, given by Prof. Patty Kostkova and Prof. João Marques-Silva.

We want to express our sincere gratitude to the Catalan Association for Artificial Intelligence (ACIA), the Research Group in Energy and Artificial Intelligence (GREiA), the Universitat de Lleida (UdL), the Universitat de Girona (UdG) and the Universitat Rovira i Virgili (URV) for their support.

Mateu Villaret, Universitat de Girona
Aïda Valls, Universitat Rovira i Virgili
Teresa Alsinet, Universitat de Lleida
Cèsar Fernández, Universitat de Lleida

Universitat de Lleida (Lleida), October 2021

This page intentionally left blank

Conference Organization

The CCIA 2021 conference was organized by the Associació Catalana d'Intel·ligència Artificial (ACIA) and the Research Group in Energy and Artificial Intelligence (GREiA) of the Universitat de Lleida (UdL).

General Chair

Aïda Valls, Universitat Rovira i Virgili

Scientific Chair

Mateu Villaret, Universitat de Girona

Local Organizing Chairs

Teresa Alsinet, Universitat de Lleida
Cèsar Fernández, Universitat de Lleida

Local Organizing Committee

Josep Argelich, Universitat de Lleida
Ramon Béjar, Universitat de Lleida
Daniel Gibert, Universitat de Lleida
Carles Mateu, Universitat de Lleida
Jordi Planes, Universitat de Lleida

Scientific Committee

Núria Agell (ESADE-URL)
Isabel Aguiló (UIB)
Guillem Alenyà (IRI-CSIC-UPC)
René Alquézar (UPC)
Cecilio Angulo (UPC)
Javier Antich (UIB)
Josep Lluís Arcos (IIIA-CSIC)
Josep Argelich (UdL)
Eva Armengol (IIIA-CSIC)
Federico Barber (UPV)
Ramón Béjar (UdL)
Lluís Belanche (UPC)
Ester Bernadó (TecnoCampus)
Christian Blum (IIIA-CSIC)
Francisco Bonnín (SRV-UIB)
Vicent Botti (UPV)
Antoni Burguera (UIB)
Carlos Carrascosa (UPV)
Gustavo Casañ (RobIn Lab, UJI)
Jesús Cerquides (IIIA-CSIC)
Hubie Chen (University of London)
Jordi Coll (Aix-Marseille Université)
Dante Conti (Univ. Est. de Campinas)
Ulises Cortés (UPC)
Vicent Costa (IIIA-CSIC)
Pilar Dellunde (UAB)
Didier Dubois (IRIT-CNRS)
Joan Espasa (University of St. Andrews)
Vlad Estivill (UPF)
Zoe Falomir (UJI)
Francesc J. Ferri (UV)
Pere García (IIIA-CSIC)
Emilio García (UIB)
Ricard Gavaldà (UPC)
Héctor Geffner (ICREA-UPF)
Karina Gibert (UPC)
Jesús Giráldez (UGR)
Lluís Godo (IIIA-CSIC)

Elisabet Golobardes (URL)
Manuel González (UIB)
Francisco Grimaldo (UV)
José Guerrero (UIB)
José M. Iñesta (UA)
Anders Jonsson (UPF)
Vicente Julián (UPV)
Jordi Levy (IIIA-CSIC)
Xavier Lladó (UdG)
Beatriz López (UdG)
Emilia López (UV)
Maite López (UB)
Felip Manyà (IIIA-CSIC)
Pere Martí (UVIC)
Sebastià Massanet (UIB)
Carles Mateu (UdL)
Joaquim Meléndez (UdG)
Pedro Meseguer (IIIA-CSIC)
Antonio Morales (UJI)
Antonio Moreno (URV)
Lledó Museros (UJI)
Ángela Nebot (UPC)
Jordi Nin (ESADE-URL)
Carles Noguera (UTIA-CAS)
Pablo Noriega (IIIA-CSIC)
Eva Onaindia (UPV)
Jordi Planes (UdL)
Enric Plaza (IIIA-CSIC)
Eloi Puertas (UB)
Oriol Pujol (UB)
Josep Puyol (IIIA-CSIC)
Gabriel Recatalà (UJI)
David Riaño (URV)
Juan Vicente Riera (UIB)
Andrea Rizzoli (IDSIA, SUPSI)
Ricardo Oscar Rodríguez (UBA)
Horacio Rodríguez (UPC)
Juan Antonio Rodríguez (IIIA-CSIC)
Jordi Sabater (IIIA-CSIC)
Maria Salamó (UB)
Miquel Sànchez (UPC)
Ismael Sanz (UJI)
Marco Schorlemmer (IIIA-CSIC)
Agustí Solanas (URV)
Josep Suy (UdG)
Carme Torras (IRI-CSIC-UPC)
Joaquín Torres (UJI)
Aïda Valls (URV)
Maria Vanrell (CVC-UAB)
Xavier Varona (UIB)
Alfredo Vellido (UPC)
Mateu Villaret (UdG)
Jordi Vitrià (UB, CVC)
Leo Wanner (ICREA-UPF)
Franz Wotawa (TU Graz)

Additional Referees

Giorgios Athanasiou (IIIA-CSIC)
Manel Rodríguez (IIIA-CSIC)
Borja Sánchez (IIIA-CSIC)

Organizing Institutions

1996 acia 2019

research group GREia

Universitat de Lleida

UNIVERSITAT ROVIRA i VIRGILI

Universitat de Girona

Contents

Invited Talks

Combinatorial Problem Solving and Logics for Artificial Intelligence

Sentiment Analysis and Text Analysis

Data Science and Decision Support Systems

Machine Learning

Computer Vision

Explainability and Argumentation

This page intentionally left blank

Invited Talks

This page intentionally left blank

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210108

Digital Public Health Technologies and Social Media in Global Emergencies

Patty KOSTKOVA [1]
UCL Centre for Digital Public Health in Emergencies (dPHE)
UCL, London, United Kingdom

Keywords. Digital Health, Emergencies prevention, Community Surveillance, Gamification

In this keynote presentation we will outline the issues surrounding the dissemination of medical evidence and understanding public information needs from Internet searches. Educational effectiveness of serious games highlights the opportunity mobile technology brought to training, community engagement and behaviour change. Social networking with increasing amount of user-generated content from social media and participatory surveillance systems provide readily available source of real-time monitoring and epidemic intelligence to prevent global emergencies such as COVID-19.

Drawing from award winning initiatives we will demonstrate the opportunity for crowdsourcing for community surveillance to combat the zika virus in Brazil, social media use for vaccination campaigns, early warning for epidemics and support for citizens in COVID-19 lockdown through a journaling app, serious mobile games to increasing resilience in perinatal women in Nepal, and decision support tools for antimicrobial prescribing in Nigeria.

[1] Corresponding Author: Patty Kostkova, p.kostkova@ucl.ac.uk

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210109

Automated Reasoning in Explainable AI

João MARQUES-SILVA [1]
IRIT, CNRS, Toulouse, France

Keywords. Decision systems, Machine Learning, Explainable AI

The envisioned applications of machine learning (ML) in high-risk and safety-critical applications hinge on systems that are robust in their operation and that can be trusted. Automated reasoning offers the solution to ensure robustness and to guarantee trust. This talk overviews recent efforts on applying automated reasoning tools in explaining black-box (and so non-interpretable) ML models [6], and relates such efforts with past work on reasoning about inconsistent logic formulas [11]. Moreover, the talk details the computation of rigorous explanations of black-box models, and how these serve for assessing the quality of widely used heuristic explanation approaches. The talk also covers important properties of rigorous explanations, including duality relationships between different kinds of explanations [7,5,4]. Finally, the talk briefly overviews ongoing work on mapping practical efficient [8,3] but also tractable explainability [9,10,2,1].

References

[1] Martinc C. Cooper and Joao Marques-Silva. On the tractability of explaining decisions of classifiers. In *CP*, October 2021.

[2] Xuanxiang Huang, Yacine Izza, Alexey Ignatiev, and Joao Marques-Silva. On efficiently explaining graph-based classifiers. In *KR*, November 2021.

[3] Alexey Ignatiev and Joao Marques-Silva. SAT-based rigorous explanations for decision lists. In *SAT*, 2021.

[4] Alexey Ignatiev, Nina Narodytska, Nicholas Asher, and João Marques-Silva. From contrastive to abductive explanations and back again. In *AI*IA*, pages 335–355, 2020.

[5] Alexey Ignatiev, Nina Narodytska, Nicholas Asher, and Joao Marques-Silva. From contrastive to abductive explanations and back again. In *KR*, November 2021.

[6] Alexey Ignatiev, Nina Narodytska, and João Marques-Silva. Abduction-based explanations for machine learning models. In *AAAI*, pages 1511–1519, 2019.

[7] Alexey Ignatiev, Nina Narodytska, and João Marques-Silva. On relating explanations and adversarial examples. In *NeurIPS*, 2019.

[8] Yacine Izza and Joao Marques-Silva. On explaining random forests with SAT. In *IJCAI*, August 2021.

[9] João Marques-Silva, Thomas Gerspacher, Martin C. Cooper, Alexey Ignatiev, and Nina Narodytska. Explaining naive bayes and other linear classifiers with polynomial time and delay. In *NeurIPS*, 2020.

[10] João Marques-Silva, Thomas Gerspacher, Martin C. Cooper, Alexey Ignatiev, and Nina Narodytska. Explanations for monotonic classifiers. In *ICML*, pages 7469–7479, 2021.

[11] João Marques-Silva and Carlos Mencía. Reasoning about inconsistent formulas. In *IJCAI*, pages 4899–4906, 2020.

[1] Corresponding Author: Joao Marques-Silva, `joao.marques-silva@irit.fr`

Combinatorial Problem Solving and Logics for Artificial Intelligence

This page intentionally left blank

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210111

On Probabilistic Logical Argumentation Based on Conditional Probability

Pilar DELLUNDE [a], Lluís GODO [b] and Amanda VIDAL [b]
[a] *Universitat Autònoma de Barcelona (UAB) and Barcelona Graduate School of Mathematics (BGSMath), Bellaterra, Spain*
[b] *Artificial Intelligence Research Institute (IIIA-CSIC), Bellaterra, Spain*

Abstract. In this paper, we introduce a framework for probabilistic logic-based argumentation inspired on the DeLP formalism and an extensive use of conditional probability. We define probabilistic arguments built from possibly inconsistent probabilistic knowledge bases and study the notions of attack, defeat and preference between these arguments. Finally, we discuss consistency properties of admissible extensions of the Dung's abstract argumentation graphs obtained from sets of probabilistic arguments and the attack relations between them.

Keywords. Probabilistic Argumentation; Logic-based Argumentation; Defeasible Knowledge; Inconsistent Probabilistic Knowledge Bases.

1. Introduction

In many scenarios, one has to deal with both uncertain and inconsistent information. Argumentation systems [4] have shown to be very suitable tools to reason with inconsistent information. In the literature, there have been a number of approaches to combine different theories of argumentation, both abstract and instantiated, with probability theory and other uncertainty models in order to allow for a more fine-grained reasoning when arguments involve uncertain information. See for instance [1,9,15,21,11,12,13,2,14,19,6].

According to [11], two main approaches can be distinguished. The epistemic approach takes the stance that the uncertainty is within the instantiated arguments (probabilities are used for capturing the strength of an argument, given the uncertainty concerning the truth of its premises or the reliability of its inferences, see for instance [20]). In the constellation approach, usually in the frame of abstract argumentation, the uncertainty is about the arguments themselves (probabilities are used for expressing uncertainty about the acceptance of the argument by some arguing agent, see for instance [11,12]). In contrast to [11], but similarly to [19], in this paper we consider logic-based arguments $A = (\mathit{support}; \mathit{conclusion})$, where *support* and *conclusion* are logical propositions, pervaded with uncertainty due a non-conclusive conditional link between their supports and their conclusions. In such a case, it is very reasonable to supplement the argument representation with a quantification α of how certain *conclusion* can be claimed to hold whenever *support* is known to hold [17], leading to represent an arguments as triples $A = (\mathit{support}; \mathit{conclusion} : \alpha)$. A very natural choice is to interpret α as some parameter related to the conditional probability $P(\mathit{conclusion} \mid \mathit{support})$. In this paper we will consider α to be a probability interval $[c_1, c_2]$, meaning that the argument A provides the information $P(\mathit{conclusion} \mid \mathit{support}) \in [c_1, c_2]$.

If we internalise the conditional link within the argument as a conditional formula *support* $\rightsquigarrow$ *conclusion* and arguments get more complex and need several uncertain conditionals to link the support with the conclusion, then we can attach conditional probability intervals to each of the involved conditionals, so arguments become of the form

$$A = (\Pi, \Delta = \{(\psi_1 \rightsquigarrow \varphi_1 : \beta_1), \ldots, (\psi_n \rightsquigarrow \varphi_n : \beta_n)\};\ \varphi\colon\ \alpha),$$

where Π is a finite set of factual (i.e. non conditional) premises, ψ_i, φ_i's are logical propositions, β_i's are probability intervals, and α is a probability interval with which φ can be logically entailed from Π and Δ. In fact, this type of arguments can be seen as a probabilistic generalization of those at work in the Defeasible Logic Programming argumentation framework (DeLP) [10].

In this paper we extend our preliminary work in [7]. First of all, we consider a more general language to build arguments, allowing conditionals (or rules) to have arbitrary propositional formulas as antecedents and consequents, while in [7] only conjunction of literals and literals were allowed. Second, we allow to attach intervals to conditionals for the corresponding conditional probabilities rather than only lower bounds as in [7]. Finally, in this paper we start the study of Dung's abstract argumentation systems associated to our probabilistic argumentation systems, and in particular the status of Caminada and Amgoud's rationality postulates [5] for the acceptability semantics based on complete extensions.

This paper is structured as follows. In Section 2 we introduce the notions about logic and probability necessary for the rest of the paper. In Section 3 we present our framework of probabilistic argumentation based on conditional probabilities, with the notion of probabilistic arguments and the attack and defeat relations among them. In Section 5 we consider the abstract argumentation system associated to a set of probabilistic arguments and their attacks, and we study the status of rationality postulates for Dung's extensions-based complete semantics. We conclude the paper with some comments on future work.

2. Logic and probability

When aiming towards the definition of a formal argumentation framework, a first step is the selection of the underlying language and logical system that will govern the derivation of new knowledge from a given set of information. Let $\mathcal{L}$ be a classical propositional language built over a finite set of variables $\mathcal{V}$ and $\vdash$ be the consequence relation of classical propositional logic.

Following [16], we introduce the set of probabilistic conditionals. We let $\mathcal{L}_{P_r}$ be the set of expressions of the form $\psi \rightsquigarrow \phi$: $[c_1, c_2]$, where ψ, ϕ are formulas of $\mathcal{L}$ and $c_1 \leq c_2$ are real numbers from the unit interval $[0, 1]$. We call ϕ the *consequent* of $\psi \rightsquigarrow \phi$: $[c_1, c_2]$, and ψ its *antecedent*. We can distinguish between classical and purely probabilistic conditionals. *Classical* conditionals are either of the form $\psi \rightsquigarrow \phi$: $[1, 1]$ or of the form $\psi \rightsquigarrow \phi$: $[0, 0]$, and purely probabilistic conditionals are of the form $\psi \rightsquigarrow \phi$: $[c_1, c_2]$, with $c_1 < 1$ and $c_2 > 0$.

Let Ω stand for the (finite) set of classical truth-evaluations $e : \mathcal{L} \to \{0, 1\}$ of the formulas in $\mathcal{L}$. Probabilities on the set of formulas $\mathcal{L}$ can be introduced in the standard way, as it is done in probability logics [6], namely by defining a probability distribution on the set of interpretations Ω, and extending it to all formulas by adding up the probabilities of their models. In other words, given a probability distribution $\pi : \Omega \to [0,1]$, i.e. is such that $\sum_{e\in\Omega} \pi(e) = 1$, then π induces a probability on formulas $P : \mathcal{L} \to [0,1]$ by stipulating

$$P(\varphi) = \sum_{e\in\Omega:e(\varphi)=1} \pi(e).$$

As defined, P satisfies the classical axioms of probability measures, namely $P(\top) = 1$ and finite additivity $P(\phi \vee \psi) = P(\phi) + P(\psi)$ whenever $\phi \wedge \psi \vdash \bot$, and moreover P respects logical equivalence: if $\vdash \phi \leftrightarrow \psi$ then $P(\phi) = P(\psi)$. Conversely, for any probability P on $\mathcal{L}$, there is a probability distribution π on Ω such that π induces P as above. Given a probabilistic conditional $\psi \rightsquigarrow \phi$: $[c_1, c_2] \in \mathcal{L}_{P_r}$, we say that a probability P on $\mathcal{L}$ *satisfies* $\psi \rightsquigarrow \phi$: $[c_1, c_2]$, if either $P(\psi) = 0$, or $P(\psi) > 0$ and $P(\phi \mid \psi) := P(\phi \wedge \psi)/P(\psi) \in [c_1, c_2]$, written

$$P \models_{pr} \psi \rightsquigarrow \phi\text{: } [c_1, c_2],$$

namely, the conditional probability of ϕ given ψ is a number in the real interval $[c_1, c_2]$. Remark that the probability of a conditional $\psi \rightsquigarrow \varphi$ is interpreted as the conditional probability $P(\phi \mid \psi)$, not as the probability of the material implication $P(\neg\psi \vee \phi)$. Moreover, we say that P *satisfies* a set of probabilistic conditionals Σ, if it satisfies each expression in Σ. We will denote the set of probabilities that satisfy Σ by $PMod(\Sigma)$.

Now we introduce a consequence relation on the set of probabilistic conditionals $\mathcal{L}_{P_r}$ first defined in [16].

Definition 2.1. *For any subset of probabilistic conditionals* $\Sigma \cup \{\psi \rightsquigarrow \phi\text{: } [c_1, c_2]\} \subseteq \mathcal{L}_{P_r}$, *the conditional* $\psi \rightsquigarrow \phi$: $[c_1, c_2]$ *is a consequence of* Σ, *denoted by* $\Sigma \models_{pr} \psi \rightsquigarrow \phi\text{: } [c_1, c_2]$, *if every probability* $P \in PMod(\Sigma)$ *satisfies* $\psi \rightsquigarrow \phi$: $[c_1, c_2]$.

Notice that, so defined, $\models_{pr}$ on $\mathcal{L}_{P_r}$ is a Tarskian consequence relation, satisfying the rules of reflexivity, monotonicity and cut.

3. Using conditional probability in arguments

3.1. Knowledge bases and arguments

Usually, arguments are built from an initial knowledge base from which one can build arguments pro and against certain pieces of information non explicitly contained in the knowledge base. Our notion of knowledge base is inspired by the approach of DeLP and other argumentation systems oriented to work with not fully reliable information. The encoding of the knowledge about a given domain in these systems distinguish pieces of knowledge considered as certain and consistent (strict knowledge) and knowledge that is tentative and subject to uncertainty or inconsistency (defeasible knowledge). If probabilities are added to the pieces of uncertain knowledge, a finer separation arises, contributing to the trustworthiness and accurateness of arguments and their relations, and allowing the argumentation system to produce more detailed outputs.

The intuition is that strict knowledge about a domain is always consistent and certain information, and hence it can be implicitly used in any argument. Thus, to specify an argument, it is only needed to specify which observations or factual information are assumed, and which part of the uncertain probabilistic knowledge is based upon. In this work, we assume the strict domain knowledge to be consistent knowledge with probability equal to 1.

Definition 3.1. *For any set of propositional formulas $S \subseteq \mathcal{L}$, let the closure of S under a set of probabilistic conditionals $\Sigma \subseteq \mathcal{L}_{P_r}$ (denoted by $Cl_\Sigma(S)$) be the smallest set containing S and the consequent of any rule in Σ whose antecedent φ is such that $Cl_\Sigma(S) \vdash \varphi$, where $\vdash$ is the consequence relation of classical propositional logic.*

Inspired in Def. 2 from [18] we distinguish two different notions of consistency.

Definition 3.2 (Direct and indirect consistency)**.** *A set $S \subseteq \mathcal{L}$ is* directly consistent *iff $S \nvdash \bot$, and* indirectly consistent *with respect to a set Γ of classical probabilistic conditionals if $Cl_\Gamma(S)$, the closure of S under Γ, is directly consistent.*

Following the usual terminology in the field of argumentation, we call *strict rule* a classical expression in $\mathcal{L}_{P_r}$ (*facts* are strict rules of the form $\top \rightsquigarrow \phi$: $[1,1]$, that we will simply denote sometimes as ϕ:$[1,1]$) and *probabilistic defeasible rule* a purely probabilistic expression in $\mathcal{L}_{P_r}$. If Π is a set of strict rules, we will denote by Π_f the set of facts in Π and we will let $\Pi_r = \Pi \setminus \Pi_f$ be the set of proper strict rules. Moreover, we will also let $\overline{\Pi}_f = \{\phi \mid \top \rightsquigarrow \phi\colon [1,1] \in \Pi_f\} \subset \mathcal{L}$ the set of consequents of the facts.

Definition 3.3. *A* probabilistic knowledge base *is a pair $KB = (\Pi, \Delta)$, where Π is a finite set of consistent strict rules and Δ is a finite set of probabilistic defeasible rules.*

Example 3.4. *The following set of probabilistic (strict and defeasible) rules $\Pi \cup \Delta$ encodes generic knowledge about inferring whether a damage (a big monetary loss) is caused in a house when the evidence (in Π) is that the alarm goes off, possibly because of a burglary or a fire.*

$$\Pi = \begin{Bmatrix} alarm\colon [1,1] \\ fire \rightsquigarrow mon\_loss\colon [1,1] \end{Bmatrix}, \quad \Delta = \begin{Bmatrix} alarm \rightsquigarrow burglary\colon [0.6, 0.8] \\ alarm \rightsquigarrow fire\colon [0.4, 0.5] \\ burglary \rightsquigarrow mon\_loss\colon [0.9, 1] \\ burglary \wedge alarm \rightsquigarrow \neg mon\_loss\colon [0.8, 1] \end{Bmatrix}$$

With the tools introduced above, we propose the following notion for a probabilistic logic-based argument.

Definition 3.5 (Argument)**.** *Given a probabilistic knowledge base $KB = (\Pi, \Delta)$, a* probabilistic argument *$\mathcal{A}$ for a formula $\theta \in \mathcal{L}$, is a tuple $\mathcal{A} = (\Sigma, \theta, [c_1, c_2])$, where $\Sigma \subseteq \Delta$, and such that:*

- *Probabilistic argument consistency: $PMod(\Sigma \cup \Pi) \neq \emptyset$*
- *Logical adequacy: θ belongs to the closure of the set of consequents of Π_f under the rules of $\Sigma \cup \Pi_r$, i.e. $\theta \in Cl_{\Sigma \cup \Pi_r}(\overline{\Pi}_f)$.*

- *c_1 (resp. c_2) is the infimum (resp. the supremum) of the values $P(\theta)$ with $P \in PMod(\Pi \cup \Sigma)$. In other words,*
  $c_1 = \sup\{d_1 \in [0,1] \mid \Pi \cup \Sigma \models_{Pr} \top \rightsquigarrow \theta\colon [d_1, 1]\}$,
  $c_2 = \inf\{d_2 \in [0,1] \mid \Pi \cup \Sigma \models_{Pr} \top \rightsquigarrow \theta\colon [0, d_2]\}$.
- *Σ is minimal satisfying the above conditions.*

If $c_2 > 0.5$ we will say the argument is proper.

In the above definition we have chosen to model evidences in an argument as part of the strict knowledge, and thus as literals with probability 1, rather than events on which to compute the conditional probability of the argument conclusion θ, see e.g. [7] where both options are considered. Of course, this issue is debatable and we let a discussion for future work. Given an argument $\mathcal{A} = (\Sigma, \theta, [c_1, c_2])$ w.r.t. $KB = (\Pi, \Delta)$, we will denote by $\Pi_{\mathcal{A}}$ the set of minimal subsets of Π such that $\mathcal{A}$ is still an argument w.r.t. $KB^* = (\Pi', \Delta)$ for each $\Pi' \in \Pi_{\mathcal{A}}$. That is, $\Pi_{\mathcal{A}}$ gathers all the minimal sets of strict rules of Π really needed in the argument $\mathcal{A}$.[1]

Thus, an argument for a literal provides for both a logical and an optimal probabilistic derivation of its conclusion from its premises. In this we follow [11] in decoupling the logic and the probabilistic aspects. Note that, in a sense, the requirements of the existence of a logical derivation and of a probabilistic entailment are rather independent. For instance, if p, q, r are variables, let $\Sigma = \{p \rightsquigarrow q\colon [\alpha, 1], q \rightsquigarrow r\colon [\beta, 1]\}$. If $0 < \alpha, \beta < 1$, then we have $r \in Cn_\Sigma(p)$, but $\{\top \rightsquigarrow p\colon [1,1]\} \cup \Sigma \not\models_{Pr} \top \rightsquigarrow r\colon [\gamma, 1]$, for any $\gamma > \alpha \cdot \beta$. Conversely, $\{\top \rightsquigarrow \neg q\colon [1,1]\} \cup \Sigma \models_{Pr} \top \rightsquigarrow \neg p\colon [1,1]$, but $\neg p \notin Cn_\Sigma(\neg q)$.

Some examples of probabilistic arguments over the KB from Example 3.4 are:

$\mathcal{A}_1 = (\{alarm \rightsquigarrow burglary\colon [0.6, 0.8], burglary \rightsquigarrow mon\_loss\colon [0.9, 1]\}; mon\_loss\colon [0.54, 1])$

$\mathcal{A}_2 = (\{alarm \rightsquigarrow burglary\colon [0.6, 0.8], burglary \wedge alarm \rightsquigarrow \neg mon\_loss\colon [0.8, 1]\}; \neg mon\_loss\colon [0.48, 1])$

$\mathcal{A}_3 = (\{alarm \rightsquigarrow fire\colon [0.4, 0.5]\}; mon\_loss\colon [0.4, 0.5])$

Note that $\mathcal{A}_1$ and $\mathcal{A}_2$ are proper, while $\mathcal{A}_3$ is not.

3.2. Counter-arguments, attacks and defeats

The next step in our formalization of a probabilistic argumentation system is to introduce the notions of subargument and attack relation between arguments. The notion of subargument is the usual one.

Definition 3.6 (Subargument)**.** *Let $\mathcal{A} = (\Sigma, \theta, [c_1, c_2])$ be an argument. A subargument of $\mathcal{A}$ is an argument $\mathcal{B} = (\Sigma', \theta', [d_1, d_2])$ where $\Sigma' \subseteq \Sigma$.*

Next we identify when two probabilistic conditionals lead to an inconsistency in the context of a KB.

Definition 3.7 (Disagreement)**.** *Let $KB = (\Pi, \Delta)$ be a probabilistic knowledge base. We say that two conditionals $\psi_1 \rightsquigarrow \varphi_1\colon [c_1, c_2]$ and $\psi_2 \rightsquigarrow \varphi_2\colon [d_1, d_2]$* disagree *whenever they are probabilistically inconsistent with the strict knowledge, i.e. when $PMod(\Pi \cup \{\psi_1 \rightsquigarrow \varphi_1\colon [c_1, c_2], \psi_2 \rightsquigarrow \varphi_2\colon [d_1, d_2]\} = \emptyset$.*

[1] It can be shown that the set $\Pi_{\mathcal{A}}$ is not necessarily a singleton.

This notion is used to define the following attack relation between arguments that is somehow more general than an undercut relation (for a general reference on this type of relations we refer to [4]).

Definition 3.8 (Attack)**.** *An argument* $\mathcal{A} = (\Sigma_1, \theta_1, [c_1, c_2])$ *attacks another argument* $\mathcal{B} = (\Sigma_2, \theta_2, [d_1, d_2])$ *at a formula* α *if there is a subargument* $\mathcal{B}' = (\Sigma_2', \alpha, [e_1, e_2])$ *of* $\mathcal{B}$ *such that* $\top \rightsquigarrow \theta_1 \colon [c_1, c_2]$ *and* $\top \rightsquigarrow \alpha \colon [e_1, e_2]$ *disagree.*

Going back to the examples above, it is clear that the argument $\mathcal{A}_2$ attacks both arguments $\mathcal{A}_1$ and $\mathcal{A}_3$, but it turns out that arguments $\mathcal{A}_1$ and $\mathcal{A}_3$ also attack each other despite of they both conclude on the same formula, the reason being that $\top \rightsquigarrow mon\_loss \colon [0.9, 1]$ and $\top \rightsquigarrow mon\_loss \colon [0.4, 0.5]$ are probabilistically inconsistent.

The next question is to specify when an attack can be deemed as effective, that is, when an attack of argument $\mathcal{A}$ to another argument $\mathcal{B}$ invalidates the latter, or in other words when an attack is actually a defeat. In our case, having probabilities in the arguments provides an additional criterion with an important role to be played. One possibility is to directly use the involved weights to decide when an argument prevails over another one. For instance, according to this criterion, argument $\mathcal{A}_1$ defeats argument $\mathcal{A}_2$. However, this seems rather counter-intuitive since, even if the derived probability in $\mathcal{A}_2$ is smaller than the one derived in $\mathcal{A}_1$, argument $\mathcal{A}_2$ is using more information than $\mathcal{A}_1$. This is in essence the specificity criterion, an approach that has been developed in non probabilistic scenarios, e.g. in [10] or [3]. Nevertheless, in case of two conflicting arguments using the same amount of information, then the comparison of the probability values surely becomes a suitable criterion to use. We start by formalizing the notion of specificity following ideas of [10].

Definition 3.9 (Activation set)**.** *Given a probabilistic knowledge base* $KB = (\Pi, \Delta)$*, an* activation set *of an argument* $\mathcal{A} = (\Sigma, \theta, [c_1, c_2])$ *is a set of formulas of the form* $Act = Ant(\Sigma \cup \Pi')$*, where* $Ant(\Sigma \cup \Pi')$ *is the set of the antecedents of all the rules in* Σ *and* Π'*, for some* $\Pi' \in \Pi_{\mathcal{A}}$*. We will denote by* $Act(\mathcal{A})$ *the set of all activation sets for* $\mathcal{A}$*.*

For instance, continuing with Example 3.4, we have that

$$\begin{aligned} Act(\mathcal{A}_1) &= \{\{alarm\}, \{burglary\}\}, \\ Act(\mathcal{A}_2) &= \{\{alarm\}, \{burglary \wedge alarm\}\}, \\ Act(\mathcal{A}_3) &= \{\{alarm\}, \{fire\}\}. \end{aligned}$$

In the following, if Γ and Γ' are two sets of formulas, we will write $\Gamma \vdash \Gamma'$ when $\Gamma \vdash \psi$ for all $\psi \in \Gamma'$. Further, if S and S' are two sets of sets of formulas, we will write $S \vdash S'$ when for all $\Gamma \in S$ there is $\Gamma' \in S'$ such that $S \vdash \Gamma'$.

Definition 3.10 (Specificity)**.** *We say that an argument* $\mathcal{A}$ *is* more specific than *another argument* $\mathcal{B}$ *when* $Act(\mathcal{A}) \vdash Act(\mathcal{B})$ *but* $Act(\mathcal{B}) \nvdash Act(\mathcal{A})$*.* $\mathcal{A}$ *and* $\mathcal{B}$ *are* equi-specific *if both* $Act(\mathcal{A}) \vdash Act(\mathcal{B})$ *and* $Act(\mathcal{B}) \vdash Act(\mathcal{A})$*, and* incomparable *whenever* $Act(\mathcal{A}) \nvdash Act(\mathcal{B})$ *and* $Act(\mathcal{B}) \nvdash Act(\mathcal{A})$*.*

It is clear that according to the above definition, concerning the arguments following Example 3.4, $\mathcal{A}_2$ is more specific than $\mathcal{A}_1$, but both are incomparable with $\mathcal{A}_3$.

In order to compare two arguments, we will then give the specificity criterion the highest priority, and if it does not produce a proper comparison, the degrees of probability will determine the strongest argument. This decision is based on the fact that probabilities of the defeasible rules inside each argument, and of the argument itself, reflect the faithfulness of each rule and indirectly, of the argument itself.

Definition 3.11 (Strength)**.** *An argument* $\mathcal{A} = (\Sigma_1, \theta_1, [c_1, c_2])$ *is* stronger *than another argument* $\mathcal{B} = (\Sigma_2, \theta_2, [d_1, d_2])$ *when* $\mathcal{A}$ *is more specific than* $\mathcal{B}$*, or otherwise when* $\mathcal{A}$ *and* $\mathcal{B}$ *are equi-specific or incomparable and* $c_1 > d_1$*.*

From the previous notions of the attack and comparative strength, it is now natural to formalize when indeed an argument defeats or rebuts another argument.

Definition 3.12 (Defeat)**.** *An argument* $\mathcal{A} = (\Sigma_1, \theta_1, [c_1, c_2])$ defeats *another argument* $\mathcal{B} = (\Sigma_2, \theta_2, [d_1, d_2])$ *when* $\mathcal{A}$ *attacks* $\mathcal{B}$ *on a subargument* $\mathcal{B} = (\Sigma_2', \alpha, [e_1, e_2])$ *and* $\mathcal{A}$ *is stronger than* $\mathcal{B}'$*.*

Continuing with Example 3.4, we have that $\mathcal{A}_2$ defeats $\mathcal{A}_1$ because $\mathcal{A}_2$ is more specific than $\mathcal{A}_1$, and $\mathcal{A}_2$ defeats $\mathcal{A}_3$ because $\mathcal{A}_2$ is not comparable with $\mathcal{A}_3$ and its probability lower bound is greater than that of $\mathcal{A}_3$. On the other hand $\mathcal{A}_1$ defeats $\mathcal{A}_3$ too, since they are also non-comparable and the probability lower bound of $\mathcal{A}_1$ is greater than the probability lower bound of $\mathcal{A}_3$.

4. Probabilistic Abstract Argumentation Frameworks

In this section we study Dung's abstract argumentation systems associated to our probabilistic argumentation systems. Namely, to any probabilistic conditional knowledge base $KB = (\Pi, \Delta)$ and set $\mathfrak{P}$ of *proper* probabilistic arguments over KB,[2] we can associate an abstract argumentation system $\langle \mathfrak{P}, R \rangle$, where R is the attack relation of Def. 3.8 on the set $\mathfrak{P}$. This is an argument graph in the sense of Dung's abstract argumentation [8].

In order to decide whether to accept a set of arguments, one can consider whether this set is conflict-free, admissible, or more in general if it fits with any of extension-based semantics proposed in the literature (complete, grounded, preferred, stable, etc). In [5] the authors define three principles, called *rationality postulates*, that can be used to identify some important properties of closure and consistency of argumentation systems. In this section, we prove that, in general, only one of these postulates is satisfied in our system. In the final section we propose some ideas in order to improve our system in a way that can satisfy the other two postulates.

Following [5], we restrict our language assuming that in the expressions of the form $\psi \rightsquigarrow \phi \colon [c_1, c_2]$, ψ can be only a literal, and ϕ a conjunction of literals. We recall now some basic notions of abstract argumentation theory. First we introduce the notion of extension under Dung's standard semantics, and then the notion of conflict-free and complete extension.

Definition 4.1 (Extension)**.** *Given an abstract argumentation framework* $\langle \mathfrak{P}, R \rangle$*, an* extension *of* $\langle \mathfrak{P}, R \rangle$ *is a subset of arguments* $E \subseteq \mathfrak{P}$*. Moreover, it is said that an extension* E defends *an argument* $\mathcal{A}$*, if for every argument* $\mathcal{B}$ *that attacks* $\mathcal{A}$*, there is an argument* $C \in E$ *that attacks* $\mathcal{B}$*.*

[2] Such a set $\mathfrak{P}$ somehow corresponds to what Hunter calls a *epistemic* extension in [11].

Given an argument $\mathcal{A}$ we denote by $Conc(\mathcal{A})$ the conclusion of $\mathcal{A}$. For instance, in case of a probabilistic argument $\mathcal{A} = (\Sigma, l, [c_1, c_2])$, $Conc(\mathcal{A}) = l$ is a literal. For any extension E, we denote by $Concs(E)$ the set of conclusions of the arguments in E.

Definition 4.2 (Conflict-free and complete extension)**.** *Given an abstract argumentation framework $\langle\mathfrak{P}, R\rangle$, it is said that an extension of $\langle\mathfrak{P}, R\rangle$, $E \subseteq \mathfrak{P}$ is* conflict-free *if there are not arguments $\mathcal{A}, \mathcal{B} \in E$ such that $\mathcal{A}R\mathcal{B}$, that is, such that $\mathcal{A}$ attacks $\mathcal{B}$. Moreover, it is said that E is* complete *if it is conflict-free and E defends all its arguments.*

Now we introduce the rationality postulates of [5]. The idea is to define postulates not only for each individual extension, but also for the set of overall justified conclusions, that is, the set called *Output* in [5], defined as follows: given the set of all extensions $E_1, \ldots, E_n$ under a given abstract semantics, $Output = \bigcap_{i \le n} Concs(E_i)$.

In what follows $\mathfrak{P}$ will denote a set of probabilistic arguments over a probabilistic conditional knowledge base $KB = (\Pi, \Delta)$, and $\langle\mathfrak{P}, R\rangle$ will stand for the associated abstract argumentation system, as described at the beginning of this section.

Definition 4.3 (Cf. [5])**.** *Let $E_1, \ldots, E_n$ be the set of all extensions of $\langle\mathfrak{P}, R\rangle$ under a given abstract semantics. Then we have the following definitions:*

Postulate 1: $\langle\mathfrak{P}, R\rangle$ satisfies Closure *if*

1. *$Concs(E_i) = Cl_\Pi(Concs(E_i))$, for each $i \le n$.*
2. *$Output = Cl_\Pi(Output)$.*

Postulate 2: $\langle\mathfrak{P}, R\rangle$ satisfies Direct Consistency *iff*

1. *$Concs(E_i)$ is consistent, for each $i \le n$.*
2. *Output is consistent.*

Postulate 3: $\langle\mathfrak{P}, R\rangle$ satisfies Indirect Consistency *iff*

1. *$Cl_\Pi(Concs(E_i))$ is consistent, for each $i \le n$.*
2. *$Cl_\Pi(Output)$ is consistent.*

Next we show that our system satisfies Postulate 2, but not in general Postulates 1 and 3. Before we prove some basic properties regarding closure under subarguments and direct consistency of complete extensions.

Proposition 4.4. *For every complete extension E of $\langle\mathfrak{P}, R\rangle$, any argument $\mathcal{A} \in E$ and subargument $\mathcal{A}'$ of $\mathcal{A}$, we have that $\mathcal{A}' \in E$.*

Proof. Assume, searching for a contradiction, that $\mathcal{A} \in E$, $\mathcal{A}'$ is a subargument of $\mathcal{A}'$ but $\mathcal{A}' \notin E$. Then, since E is complete, either $E \cup \{\mathcal{A}`\}$ is not conflict-free, or $E$ does not defend $\mathcal{A}'$. In the first case, if $E \cup \{\mathcal{A}`\}$ is not conflict-free, there is a $\mathcal{B} \in E$ such that either $\mathcal{B}$ attacks $\mathcal{A}'$, or $\mathcal{A}'$ attacks $\mathcal{B}$. If $\mathcal{B}$ attacks $\mathcal{A}'$, by Def. 3.8, $\mathcal{B}$ attacks also $\mathcal{A}$, contradicting the fact that E is conflict-free. If $\mathcal{A}'$ attacks $\mathcal{B}$, since E is complete, E defends $\mathcal{B}$, and thus, there is a $C \in E$ that attacks $\mathcal{A}'$. Therefore, by Def. 3.8, C attacks also $\mathcal{A}$, contradicting the fact that E is conflict-free. In the second case, if E does not defend $\mathcal{A}'$, there is a $\mathcal{D} \in \mathfrak{P}$ that attacks $\mathcal{A}'$ and no $C \in E$ attacks $\mathcal{D}$. Then, by Def. 3.8, $\mathcal{D}$ attacks also $\mathcal{A}$ and E does not defend $\mathcal{A}$, which is a contradiction, because E is complete.

Since in both cases we have reached a contradiction, we can conclude that $\mathcal{A}' \in E$, i.e., E is closed under subarguments. □

Proposition 4.5. *For every conflict-free extension E of $\langle \mathfrak{P}, R\rangle$, $Concs(E)$ is directly consistent.*

Proof. Assume, searching for a contradiction, that $Concs(E)$ is not directly consistent, i.e. by Def. 3.2, $Concs(E) \vdash \perp$. Since $Concs(E)$ is a set of literals, there are arguments $\mathcal{A} = (\Sigma_1, l_1, [c_1, c_2])$, and $\mathcal{B} = (\Sigma_2, l_2, [d_1, d_2])$, with $\mathcal{A}, \mathcal{B} \in E$ and such that $l_1 = -l_2$.

Since E is conflict-free, $\mathcal{A}$ does not attack $\mathcal{B}$, and $\mathcal{B}$ does not attack $\mathcal{A}$. Thus, by Def. 3.8, $PMod(\Pi \cup \{(l_1 \mid \top)[c_1, c_2], (l_2 \mid \top)[d_1, d_2]\}) \neq \emptyset$, with $c_1, d_1 > 0.5$. Then $l_1 \neq -l_2$, contradicting our original assumption. Therefore $Concs(E)$ is directly consistent. □

Theorem 4.6. *If $\{E_1, \ldots, E_n\}$ is the set of all the extensions of $\langle \mathfrak{P}, R\rangle$ under the complete semantics, then $\langle \mathfrak{P}, R\rangle$ satisfies Postulate 2 but not necessarily Postulates 1 and 3.*

Proof. (Postulate 2) By Prop. 4.5, because for each $i \leq n$, E_i is complete, and thus conflict-free. Moreover, since the intersection of consistent sets is also consistent, we have that *Output* is also consistent.

(Postulates 1 and 3) Let $KB = (\Pi, \Delta\rangle)$ be the following probabilistic conditional knowledge base:

$$\Pi = \{\top \rightsquigarrow d\colon [1,1], \top \rightsquigarrow e\colon [1,1], \top \rightsquigarrow f\colon [1,1], a \wedge b \rightsquigarrow \neg c\colon [1,1]\}$$

$$\Delta = \{r_1 : d \rightsquigarrow a\colon [0.9,1], r_2 : e \rightsquigarrow b\colon [0.7,1], r_3 : f \rightsquigarrow c\colon [0.8,1]\}$$

Consider now the following set $\mathfrak{P}$ of proper arguments over KB:

$$\mathcal{A}_1 = (\{r_1\}, a, [0.9,1]),\ \mathcal{A}_2 = (\{r_2\}, b, [0.7,1]),$$
$$\mathcal{A}_3 = (\{r_3\}, c, [0.8,1]),\ \mathcal{A}_4 = (\{r_1, r_2\}, \neg c, [0.6,1]).$$

On this set of arguments, the relation of attack is the following: $\mathcal{A}_3$ attacks $\mathcal{A}_4$, and $\mathcal{A}_4$ attacks $\mathcal{A}_3$. Thus, the abstract argumentation system associated to this set of arguments has the two following complete extensions: $E = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3\}$ and $F = \{\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_4\}$. This is a counterexample for both Postulates 1 and 3, because E is not closed under strict rules and it is not indirectly consistent. □

5. Future work

In this paper we have proposed a framework for logic-based probabilistic argumentation where conditional expressions are qualified with conditional probability intervals, building on Lukasiewicz's setting for probabilistic logic programming with conditional constraints [16], and extending previous work [7].

There are many open problems left for future work. Here we mention some of them. First of all, although the underlying language based on conditionals is quite general, we could consider a more powerful logic to reason with those conditionals. We also need to consider and study possible alternatives to the notion of subargument, for instance in the line of [2] where a more refined notion is at work, and to the attack and defeat relations with possibly more suitable comparison criteria. Also observe that interval-valued probabilistic conditionals can be equivalently expressed by pairs of two lower bound-valued conditionals, which would be an interesting research line. Referring to this latter issue, and its influence in the fulfilment of rationality postulates by the associated abstract argumentation systems studied in the previous section, a promising prospect seems to consider a notion of collective conflict, also similarly to [2]. Finally, we can also mention the question of studying whether the probabilities involved in the arguments could allow for gradual notions of attack and acceptability.

Acknowledgments The authors are grateful to the reviewers and acknowledge partial support by the H2020-MSCA-RISE-2020 project MOSAIC (Grant Agreement 101007627) and the Spanish project ISINC (PID2019-111544GB-C21).

References

[1] T. Alsinet, C.I. Chesñevar, L. Godo, and G.R. Simari. A logic programming framework for possibilistic argumentation: Formalization and logical properties. *Fuzzy Sets and Systems*, 159(10):1208-1228, 2008.

[2] T. Alsinet, R. Béjar, L. Godo, F. Guitart. RP-DeLP: a weighted defeasible argumentation framework based on a recursive semantics. *J. Log. Comput.* 26(4): 1315-1360, 2016.

[3] D. Bamber, I.R. Goodman, and H.T. Nguyen. Robust reasoning with rules that have exceptions: From second-order probability to argumentation via upper envelopes of probability and possibility plus directed graphs. *Ann Math Artif Intell.*, 45:83–171, 2005.

[4] P. Baroni, D.M. Gabbay, M. Giacomin, and L. van der Torre (eds.) Handbook of Formal Argumentation. College Publications, 2018.

[5] M. Caminada and L. Amgoud. On the evaluation of argumentation formalisms. *Artificial Intelligence*, 171:286–310, 2007.

[6] F. Cerutti and M. Thimm. A general approach to reasoning with probabilities. *International Journal of Approximate Reasoning*, 111:35–50, 2019.

[7] P. Dellunde, L. Godo, and A. Vidal. Probabilistic Argumentation: An Approach Based on Conditional Probability -A Preliminary Report. In Proc. JELIA'21, *LNAI*, vol 12678, 25-32, 2021.

[8] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and n–person games. *Artificial Intelligence*, 77(2):321–357, 1995.

[9] P. M. Dung and P. M. Thang. Towards probabilistic argumentation for jury-based dispute resolution. In P. Baroni et al. (eds.), *Proc. of COMMA 2010*, vol. 216 of FAIA, pages 171–182. IOS Press, 2010.

[10] A. Garcia and G. Simari. Defeasible logic programming: an argumentative approach. *Theory and Practice of Logic Programming*, 4(1-2):95–138, 01 2004.

[11] A. Hunter. A probabilistic approach to modelling uncertain logical arguments. *International Journal of Approximate Reasoning*, 54(1):47–81, 2013.

[12] A. Hunter. Probabilistic qualification of attack in abstract argumentation. *IJAR*, 55(2):607–638, 2014.

[13] A. Hunter and M. Thimm. On partial information and contradictions in probabilistic abstract argumentation. In C. Baral et al. (eds.), *Proc. of KR 2016*, pages 53–62. AAAI Press, 2016.

[14] A. Hunter and M. Thimm. Probabilistic reasoning with abstract argumentation frameworks. *J. Artif. Intell. Res.*, 59:565–611, 2017.

[15] H. Li, N. Oren, and T. J. Norman. Probabilistic argumentation frameworks. In S. Modgil et al. (eds.), *Theory and Applications of Formal Argumentation - First International Workshop, TAFA 2011, Revised Selected Papers*, Vol. 7132 of *LNCS*, pages 1–16. Springer, 2011.

[16] T. Lukasiewicz. Probabilistic logic programming with conditional constraints. ACM Transactions on Computational Logic, 2(3): 289-339 (2001)

[17] J.L. Pollock. Justification and defeat. *Artificial Intelligence*, 67:377–408, 1994.

[18] H. Prakken. Historical overview of formal argumentation. In P. Baroni et al. (eds.) *Handbook of Formal Argumentation*, volume 1, pages 73–141. College Publications, 2018.

[19] H. Prakken. Probabilistic strength of arguments with structure. In M. Thielscher et al. (eds.), Francesca Toni, and Frank Wolter, editors, Proc. of KR 2018, pages 158–167. AAAI Press, 2018.

[20] S. Timmer, J.J.Ch. Meyer, H. Prakken, S. Renooij, and B. Verheij. A two-phase method for extracting explanatory arguments from bayesian networks. *Int. J. Approx. Reason.*, 80:475–494, 2017.

[21] B. Verheij. Jumping to conclusions: a logico-probabilistic foundation for defeasible rule-based arguments. In A. Herzig et al. (eds.), *Proc. of JELIA 2012*, LNAI, vol. 7519, 411–423, 2012.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210112

Application of CMSA to the Minimum Positive Influence Dominating Set Problem

Mehmet Anıl AKBAY [a,1] and Christian BLUM [a]
[a] *Artificial Intelligence Research Institute (IIIA-CSIC), Campus UAB, Bellaterra, Spain*

Abstract. Construct, Merge, Solve & Adapt (CMSA) is a recently developed algorithm for solving combinatorial optimization problems. It combines heuristic elements, such as the probabilistic generation of solutions, with an exact solver that is iteratively applied to sub-instances of the tackled problem instance. In this paper, we present the application of CMSA to an NP-hard problem from the family of dominating set problems in undirected graphs. More specifically, the application in this paper concerns the minimum positive influence dominating set problem, which has applications in social networks. The obtained results show that CMSA outperforms the current state-of-the-art metaheuristics from the literature. Moreover, when instances of small and medium size are concerned CMSA finds many of the optimal solutions provided by CPLEX, while it clearly outperforms CPLEX in the context of the four largest, respectively more complicated, problem instances.

Keywords. Construct, merge, solve & adapt, minimum positive influence dominating set, hybrid metaheuristics

1. Introduction

When faced with a hard combinatorial optimization problem, the related literature generally offers both exact and approximate techniques for solving the problem. Exact techniques guarantee to find an optimal solution to a given problem instance in bounded computation time. Therefore, instances up to a problem-specific size and/or difficulty are usually solved by using an exact technique. Hereby, the term exact technique might refer to a specialized algorithm or to a general-purpose tool such as, for example, an integer linear programming (ILP) solver. Examples for ILP solvers include CPLEX and Gurobi, just to name the currently most powerful ones. The computation time required by an exact technique generally starts to explode when reaching a problem-specific instance size and/or difficulty. When this happens researchers and practitioners usually resort to using approximate techniques for obtaining solutions to their problem. Examples range from simple greedy heuristics to more sophisticated metaheuristics [2]. In order to take profit from the valuable optimization expertise that has gone into the development of exact optimization tools such as CPLEX and Gurobi, in the last two decades some researchers have focused on the development of algorithms that allow to take profit from

[1] Corresponding Author; E-mail: makbay@iiia.csic.es

these tools even in the context of problem instances that are too large to be solved directly by them. Algorithms from this line of research are called *hybrid metaheuristics* or *matheuristics* [5]. Prominent examples include algorithms such as large neighborhood search (LNS) [13] and construct, merge, solve & adapt (CMSA) [1].

In this paper we demonstrate the application of CMSA to the so-called minimum positive influence dominating set (MPIDS) problem [16,17]. The MPIDS problem is an NP-hard combinatorial optimization problem with applications in social networks. Each vertex in such a network represents an individual—that is, a person—and edges indicate relationships, respectively interactions, between those individuals. The background of the MPIDS problem is that information propagated in social networks can have a significant, either positive or negative, impact on the respective parts of the society. From social norms theory it is known that the behavior of individuals can be affected by the perception of others' thoughts and behaviors [6]. This makes it possible to exploit the relationships among people in social networks in order to obtain great benefits for both the economy and society. The aim of the MPIDS problem is to identify a small subset of influential individuals (or key individuals) in order to accelerate the spread of positive influence in a social network [10,7]. Alternative applications of the MPIDS problem can be found in e-learning software [18], online business [14], drinking, smoking, and other drug-related problems [16].

The remaining part of this paper is organized as follows. A technical description of the MPIDS problem, together with a standard ILP model, is provided in Section 2. The application of CMSA to the MPIDS problem is described in Section 3. Finally, an experimental evaluation, including a comparison to the state-of-the-art from the literature, can be found in Section 4, and conclusions as well as an outlook to future lines of research are provided in Section 5.

2. MPIDS problem

In technical terms, the MPIDS problem can be described as follows. Given a simple that does not contain any loops and parallel edges, connected, undirected graph $G = (V, E)$, the problem requires to find a subset S^* of V of minimum cardinality such that the following two conditions are fulfilled:

1. S^* is a dominating set of G. Remember that a subset $S \subseteq V$ of the vertices of an undirected graph G is called a dominating set, if and only if each vertex $v \in V$ forms either part of S or has at least one neighbor that forms part of S.
2. At least half of the neighbors of each vertex $v \in V$ form part of S^*.

Most of the research efforts concerning the MPIDS problem have been focused on greedy heuristics [17,15,4,12,3]. Moreover, a swarm intelligence based algorithm [9] and an ILP-based memetic algorithm [8] were presented in the literature. The latter one is currently state-of-the-art for the MPIDS problem.

Note that the MPIDS problem can easily be stated in terms of an ILP. The model is based on a binary variable x_i for each vertex $v_i \in V$.

$$\textbf{Minimize} \quad \sum_{i=1}^{n} x_i \tag{1}$$

$$\textbf{Subject to} \quad \sum_{v_j \in N(v_i)} x_j \geq \left\lceil \frac{deg(v_i)}{2} \right\rceil \quad \forall v_i \in V \tag{2}$$

$$x_i \in \{0,1\} \tag{3}$$

Hereby, $N(v_i)$ is the neighborhood of v_i in input graph G, and $deg(v_i)$ is the degree of vertex v_i, where $deg(v_i) := |N(v_i)|$. Equation (2) ensures that a feasible solution contains at least half of the neighbors of each vertex $v_i \in V$. In the context of the CMSA algorithm outlined in the next section, the objective function value $f(S)$ of a feasible solution $S \subseteq V$ is $f(S) := |S|$. Note that $S := V$ is a trivial solution to the problem.

3. The CMSA Algorithm

The general structure of the CMSA developed for the MPIDS problem is presented in Algorithm 1. The algorithm starts by taking an instance represented by a simple, connected, undirected graph $G = (V, E)$ as input. $S \subseteq V$ denotes a feasible MPIDS solution of G. At first, the best-so-far solution S_{bsf} is initialized to the trivial solution V. Then, two vector data structures, called $age_0[]$ and $age_1[]$, are initialized to value -1 for all $v \in V$; see line 4. Note that values $age_0[v]$ and $age_1[v]$ may range between -1 and a fixed positive integer value called age_{max}. Hereby, age_{max} is one of CMSA's important parameters. These data structures are modified in two parts of the algorithm, namely (1) on the basis of the generated solutions and (2) on the basis of solving the so-called sub-instance. This is explained in detail below. After the initialization of data structures $age_0[]$ and $age_1[]$, a pre-processing procedure from [3] is applied to determine the set S_{par} of vertices that must form part of an optimal solution; see line 5. At each iteration, the algorithm probabilistically generates n_a solutions by applying function ProbablisticSolutionGeneration(S_{par}) in line 8. In order to probabilistically generate a solution S, a recent greedy algorithm from [3] is applied in a probabilistic way. This is also explained in Section 3.1. Afterwards, a sub-instance is generated on the basis of the current values in data structures $age_0[]$ and $age_1[]$.For a detailed explanation of the data structures used for defining a sub-instance, see Section 3.2. This sub-instance is then solved by CPLEX with a CPU time limit of t_{ILP} seconds by applying function SolveSubinstance($age_0[], age_1[], t_{\text{ILP}}$) ; see line 14. The result is a solution S_{opt}, which is a solution to both the sub-instance and the original problem instance. Note that t_{ILP} seconds may, or may not, be enough time for CPLEX to solve the sub-instance to optimality. In case t_{ILP} is not enough time, S_{opt} is a sub-optimal solution to the sub-instance. Next, the best-of-far solution S_{bsf} is updated with S_{opt} in case $f(S_{opt}) < f(S_{bsf})$; see line 15. Finally, the values of data structures $age_0[]$ and $age_1[]$ are modified on the basis of solution S_{opt} as shown in Section 3.3 in detail. The algorithm stops once the CPU time limit is reached. In the following, the remaining parts of the algorithm are outlined in more detail.

3.1. Probabilistic construction of solutions

Function ProbablisticSolutionGeneration(S_{par}) generates a valid solution S as follows. First, S is initialized to S_{par}. Note that, by initializing all solutions to be constructed by

Algorithm 1 CMSA for the MPIDS problem

```
input: a problem instance $G = (V, E)$
parameters: $n_a$, $d_{\text{rate}}$, $l_{\text{size}}$, $\text{age}_{\max}$, and $t_{\text{ILP}}$
$S_{bsf} := V$
$age_0[v] := -1$ and $age_1[v] := -1$ for all $v \in V$
$S_{par} :=$ PreProcessing($G$)
while CPU time limit not reached do
    for $k := 1, \ldots, n_a$ do
        $S :=$ ProbablisticSolutionGeneration($S_{par}$)
        for all $v \in V$ do
            if $v \in S$ and $age_1[v] = -1$ then $age_1[v] := 0$
            if $v \notin S$ and $age_0[v] = -1$ then $age_0[v] := 0$
        end for
    end for
    $S_{opt} :=$ SolveSubinstance($age_0[], age_1[], t_{\text{ILP}}$)
    if $f(S_{opt}) < f(S_{bsf})$ then $S_{bsf} := S_{opt}$
    Adapt($age_0[], age_1[], S_{bsf}, \text{age}_{\max}$)
end while
return: $S_{bsf}$, the best solution found by the algorithm
```

S_{par}, the algorithm's performance is enhanced because the construction of solutions is accelerated. After this initialization, the set U of uncovered vertices with respect to S is determined. In this context, note that a vertex $v \in V$ is called *covered* with respect to a (partial) solution S if and only if at least half of its neighbors form part of S. In the opposite case, v is defined as *uncovered*. The following steps are then repeated until no uncovered vertices are left:

1. A vertex $v \in U$ with the smallest neighborhood size ($deg(v)$) is chosen. In other words, a vertex $v \in U$ is chosen such that $deg(v) \leq deg(v')$ for all $v' \in U$.
2. Afterward, vertices are iteratively chosen from $N(v) \setminus S$ and added to S until v is covered. The minimum number of adjacent vertices ($h_S(v)$) that need to be chosen and added to S is calculated using the following equation: $h_S(v) := \lceil \frac{deg(v)}{2} \rceil - |N_S(v)|$. Here, $N_S(v)$ refers to the set of neighbors of v that form already part of solution S. In contrast to the original greedy algorithm from [3], a vertex $v_i \in N(v) \setminus S$ may either be selected in a deterministic or in a probabilistic way. For this, we utilize two important parameters, namely the determinism rate d_{rate} and the candidate list size l_{size}. At first, a candidate list L is created. This list includes all the vertices $v' \in N(v) \setminus S$. Each vertex v' in L is characterized by its *cover degree*, which is the number of uncovered adjacent vertices of v'. Note that vertices in L are sorted according to a non-increasing cover degree value. Then, a uniform random number r is generated from the interval $[0, 1]$. If $r \leq d_{\text{rate}}$, the vertex with the highest cover degree is selected and added to S. Otherwise, a vertex is selected randomly from the restricted candidate list which contains the first l_{size} vertices of L. All vertices in the restricted candidate list have an equal probability $\frac{1}{l_{\text{size}}}$ of being selected.
3. The set U of uncovered vertices is re-computed.

3.2. Definition and solution of the sub-instance

Before describing the modification of data structures $age_0[]$ and $age_1[]$, we first explain how the values in these data structures are used for defining the sub-instance, which is obtained as an ILP model with additional restrictions. In other words, the sub-instance is obtained by adding additional constraints to the ILP model from Section 2. This ILP model is obtained as follows. First, the values $age_0[]$ and $age_1[]$ are used for splitting the set of vertices into three disjoint subsets: $V_{in} \subseteq V$ is the set of vertices that are forced to form part of any solution of the sub-instance. $V_{out} \subseteq V$ is the set of vertices that are excluded from any solution to the sub-instance. Finally, $V_{open} \subseteq V$ is the set of vertices that may, or may not, form part of a solution to the sub-instance.

- V_{in} contains all vertices $v \in V$ with $age_0[v_i] = -1$ and $age_1[v_i] \geq 0$.
- V_{out} contains all vertices $v \in V$ with $age_0[v_i] \geq 0$ and $age_1[v_i] = -1$.
- V_{open} contains all remaining vertices.

The corresponding ILP model is obtained by adding a constraint $x_i = 1$ for all $v_i \in V_{in}$, and a constraint $x_i = 0$ for all $v_i \in V_{out}$. After generating the restricted ILP which corresponds to the sub-instance, CPLEX is applied to the restricted ILP with a computation time limit of t_{ILP} seconds, resulting in a solution S_{opt}. Note that the more restricted a sub-instance is, the easier it is for CPLEX to derive an optimal solution to the sub-instance.

3.3. Modification of the data structures

As mentioned above, data structures $age_0[]$ and $age_1[]$ are modified (1) after the construction of a solution S (see lines 9-12 of Algorithm 1) and (2) after solving the sub-instance (line 16). Both cases are explained below.

After the construction of a solution S in line 8 of Algorithm 1, the following modifications are performed for each $v \in V$:

- If $v \in S$ and $age_1[v] = -1$, then $age_1[v] := 0$. This means that if (1) v forms part of S and if (2) v is currently excluded from forming part of solutions to the sub-instance (due to $age_1[v] = -1$), then $age_1[v]$ is set to zero. This means that v can now be considered for the inclusion in solutions to the sub-instance.
- If, otherwise, $v \notin S$ and $age_0[v] = -1$, then $age_0[v] := 0$ This means that if (1) v does not form part of S and if (2) v is currently not excluded to form part of solutions to the sub-instance, it may now be considered for exclusion.

Next we describe the modification of the data structures after solving the current sub-instance, that is, after generating a solution S_{opt} in the current iteration. This modification is done in function $\mathsf{Adapt}(age_0[], age_1[], S_{bsf}, \text{age}_{\max})$ (see line 16 of Algorithm 1). The working of this function is pseudo-coded in Algorithm 2. If a vertex $v \in V_{open}$ is not chosen by CPLEX for solution S_{opt}, two actions are performed: first, $age_0[v]$ is set to zero, and second, $age_1[v]$ is increased by one. In case $age_1[v]$ reaches age_{max}, $age_1[v]$ is set to its default value -1, which means that vertex v is excluded from the sub-instance in the next iteration. In other words, the vertex is transferred from set V_{open} to set V_{out} since it has not been selected by CPLEX to form part of S_{opt} during the last $\text{age}_{\max}$ iterations. Similarly, if a vertex $v \in V_{open}$ is frequently chosen by CPLEX for solution S_{opt}, it is transferred from set V_{open} to set V_{in} as described in line 9 of Algorithm 2.

Algorithm 2 Function Adapt($age_0[], age_1[], S_{bsf}, \mathrm{age}_{\max}$)

```
input: sub-instance ($C^{'}$)
for all $v \in V$ do
    if $v \in V_{open}$ then
        if $v \notin S_{bsf}$ then
            $age_0[v] := 0$ and increase $age_1[v]$ by 1
            if $age_1[v] = \mathrm{age}_{\max}$ then $age_1[v] := -1$
        else
            $age_1[v] := 0$ and increase $age_0[v]$ by 1
            if $age_0[v] = \mathrm{age}_{\max}$ then $age_0[v] := -1$
        end if
    else
        if $age_0[v] \geq 0$ then $age_0[v] := 0$
        if $age_1[v] \geq 0$ then $age_1[v] := 0$
    end if
end for
output: $C^{'}$, updated sub-instance
```

4. Experimental Evaluation

In the following we compare CMSA with the following approaches: (1) application of CPLEX 12.10 in one-threaded mode (with a computation time limit of 2 hours per problem instance); (2) IGA-PIDS, which is the currently best greedy approach from [3]; (3) HSIA, a hybrid swarm intelligence based algorithm from [9]; and (4) ILPMA, an ILP-based memetic algorithm from [8]. The experiments concerning CMSA, CPLEX and IGA-PIDS were performed on a cluster of machines with Intel® Xeon® CPU 5670 CPUs with 12 cores of 2.933 GHz and a minimum of 32 GB RAM. As in the standalone application of CPLEX, sub-instances in CMSA were solved using CPLEX 12.10 in one-threaded mode. The results for HSIA and ILPMA were taken from the respective publications. Unfortunately, they were not available for all problem instances studied in this work. CMSA, CPLEX and IGA-PIDS were applied to 17 social networks that are usually used in the literature on the MIPDS problem. These networks are of small and medium size that contain between 34 and 36692 nodes and between 788 and 198050 edges. In addition, the three algorithms were applied to 10 larger social networks from the SNAP library that contain between 37700 and 1134890 nodes and between 2289003 and 3387388 edges (`https://snap.stanford.edu/data/`).

CMSA requires well-working values for n_a (number of solution constructions per iteration), d_{rate} (determinism rate), l_{size} (candidate list size), $\text{age}_{\max}$ (upper limit for the age-values), and t_{ILP} (time limit for CPLEX per iteration). The scientific tuning software irace [11] was used for tuning these parameters. More specifically, irace was used for generating two parameter settings for CMSA: one of the 17 small/medium sized instances, and another one for the 10 large networks. Networks `CA-AstroPh`, `Email-Enron` and `socfb-Brandeis99` were used for the first tuning experiment, and networks `Amazon0312` and `Amazon0601` were used for the second one. Finally, for each of the two applications of irace the budget was fixed to 2000 algorithm runs, each

Table 1. Parameter values obtained for CMSA by tuning with irace.

Networks	n_a	d_{rate}	l_{size}	$\text{age}_{\max}$	t_{ILP}
Small/medium size	1	0.1	8	4	16
Large size	1	0.9	8	1	13

one with a time limit of 600 CPU seconds. The considered parameter value domains were as follows: $n_a \in \{1,\ldots,20\}$, $d_{\text{rate}} \in \{0.0, 0.1, 0.2, \ldots, 0.8, 0.9\}$, $l_{\text{size}} \in \{3,\ldots,10\}$, $\text{age}_{\max} \in \{1,\ldots,10\}$ and $t_{\text{ILP}} \in \{1,\ldots,30\}$ (in seconds). The obtained parameter value settings are shown in Table 1. It is worth noting that the value of parameter $\text{age}_{\max}$ decreases and the value of parameter d_{rate} increases as instance size grows to keep the sub-instance small enough to be solved by CPLEX.

While CPLEX and IGA-PIDS were applied exactly once to each of the 27 problem instances, CMSA was applied 10 times to each instance. A computation time limit of 2 hours was given to each CPLEX run, while a limit of 600 seconds was applied to each run of CMSA.

The results, in comparison to those of CPLEX (with a time limit of 2 hours per instance), IGA-PIDS [3], HSIA [9], ILPMA [8], and are shown in Table 2 (small/medium instances) and Table 3 (large instances). Both tables have the following structure. The first column contains the instance name. Columns with heading 'q' report on the quality of the best solutions found be the five approaches, and columns with heading 'avg' provide the average solution quality obtained. Furthermore, columns with heading '$\overline{t(s)}$' indicate the average computation times of ILPMA and CMSA to find the best solutions in each run. In addition, the column with heading '$\overline{t(s)}$' shows the computation time of the greedy approach IGA-PIDS. Finally, the gap (in percent) between the solution obtained by CPLEX and the best lower bound is indicated in the column with heading 'gap(%)'. Note that when the gap is zero, CPLEX was able to prove optimality. The best result for each instance is shown in bold font.

The following observations can be made. First, CPLEX performs very strongly for all small/medium size instances, and for five of the large instances. However, for the remaining five large instances it fails to find any other than the trivial solution within 2 hours of computation time. CMSA performs very comparably to CPLEX for the small/medium size instances. In one case (`CA-AstroPh`) CMSA finds a better solution than CPLEX. In five other instances it provides results that are marginally worse than those of CPLEX. Moreover, CMSA obtains the best average solution quality in the case of the 10 large problem instances. In particular, CMSA finds solutions much better than the trivial ones in the case of the five instances for which CPLEX fails. Moreover, CMSA significantly outperforms the current state-of-the-art approaches from the literature (HSIA and ILPMA).

Finally, the results also show that there is surely the need to find a way to improve CMSA for very large instances such as the last three instances of Table 3. The greedy approach IGA-PIDS outperforms CMSA in the case of these three instances.

5. Conclusions and Outlook

In this study, one of the recent hybrid metaheuristics (construct, solve, merge & adapt) was proposed to solve the minimum positive influence dominating set problem. Costruct,

Table 2.: Numerical results for small to medium size instances.

Network	CPLEX		IGA-PIDS		HSIA		ILPMA			CMSA		
	q	gap (%)	q	$\overline{t(s)}$	q	avg	q	avg	$\overline{t(s)}$	q	avg	$\overline{t(s)}$
Karate	**15**	0.00	**15**	0.0	n.a.	n.a.	**15**	**15.0**	0.03	**15**	15.00	0.0
Dolphins	**30**	0.00	31	0.0	n.a.	n.a.	**30**	**30.0**	0.13	**30**	30.00	0.011
Football	**63**	0.00	68	0.0	n.a.	n.a.	65	65.65	0.54	**63**	63.00	14.96
Jazz	**79**	0.00	81	0.0	n.a.	n.a.	n.a.	n.a.	n.a.	**79**	79.00	0.21
CA-AstroPh	6740	0.30	6953	0.031	6905	6906.6	6857	6865.45	300.41	**6736**	6739.90	539.18
CA-GrQc	**2587**	0.00	2607	0.0	2597	2598.4	2594	2596.05	45.07	**2587**	2587.00	3.17
CA-HepPh	**4718**	0.01	4817	0.015	4791	4792.4	4770	4773.85	157.43	**4718**	4718.10	183.01
CA-HepTh	**4471**	0.00	4544	0.0	4515	4516.2	4502	4506.25	107.93	**4471**	4471.00	10.89
CA-CondMat	**9584**	0.06	9748	0.015	9729	9734.0	9683	9689.6	506.37	9585	9585.60	460.42
Email-Enron	**11682**	0.00	11843	0.031	11865	11873.4	11814	11818.95	760.08	11683	11683.80	183.16
ncstrlwg2	**2994**	0.00	3010	0.015	3004	3005.4	3001	3002.85	65.69	**2994**	2994.00	14.57
actors-data	**3092**	0.24	3147	0.016	3143	3144.5	3130	3134.5	137.74	**3092**	3093.50	467.65
ego-facebook	**1973**	0.00	1975	0.078	1726[a]	1726.6[a]	1737[a]	1741.55[a]	56.91	**1973**	1973.00	59.16
socfb-Brandeis99	**1400**	1.41	1502	0.032	n.a.	n.a.	n.a.	n.a.	n.a.	1405	1408.20	377.66
socfb-nips-ego	**1398**	0.00	**1398**	0.016	n.a.	n.a.	n.a.	n.a.	n.a.	**1398**	1398.00	0.024
socfb-Mich67	**1329**	1.56	1427	0.015	n.a.	n.a.	n.a.	n.a.	n.a.	1336	1338.70	384.60
soc-gplus	**8244**	0.00	8289	0.031	n.a.	n.a.	n.a.	n.a.	n.a.	8253	8254.20	486.03
average	**3552.88**		3615.00							3554.00		

[a]: apparently, papers on HSIA and ILPMA have used a different ego-facebook instance

Table 3. Numerical results for large instances.

Network	CPLEX		IGA-PIDS		CMSA		
	q	gap (%)	q	$\overline{t(s)}$	q	avg	$\overline{t(s)}$
musae_git	**9752**	0.00	10386	1.98	10017	10027.20	586.25
loc-gowalla_edges	**67617**	0.07	69086	0.21	67931	67946.60	595.07
gemsec_facebook_artist	**15194**	1.20	16217	0.21	15405	15418.40	595.73
deezer_HR	54573	95.68	23738	0.14	**22713**	22747.80	592.43
com-youtube	**351281**	0.00	353387	2.98	352566	352587.00	593.50
com-dblp	**120492**	0.08	121940	0.31	120970	120993.60	599.43
Amazon0302	262111	97.50	134569	0.23	**130303**	130360.70	592.74
Amazon0312	400727	95.41	**180853**	0.67	183093	183103.90	282.99
Amazon0505	410236	95.19	**183114**	0.64	185230	185279.40	411.53
Amazon0601	403394	96.94	**179964**	0.66	182202	182231.30	252.42
average	209537.70		127325.40		**127043.00**		

Merge, Solve & Adapt is based on the probabilistic construction of solutions, which are used to extend the restricted sub-instance. This sub-instance is then solved by CPLEX at each iteration. Based on the results provided by CPLEX, the restricted sub-instance is modified and passed to the next iteration. Note that this procedure allows to make a beneficial use of high-performance ILP solvers such as CPLEX even in the context of problem instances that are too large for CPLEX to be applied directly.

Computational experiments were performed on 17 small/medium sized social networks and on ten larger benchmark instances from the SNAP database. The proposed approach was evaluated and compared to the state-of-the-art methods from the literature (including two metaheuristics and one greedy approach) and to the results obtained by the ILP solver CPLEX 12.10. The analysis of the results showed that construct, solve, merge & adapt outperforms the metaheuristics from the literature. Moreover, apart from the largest three problem instances, our approach outperforms the greedy approach. In comparison to CPLEX our approach obtains comparable results for small/medium sized instances, and starts to outperform CPLEX as the instance size grows.

On the negative side, we realized that the performance of our approach starts to degrade for the largest three problem instances. This means that, in these cases, even sub-instances are too large for CPLEX to be solved in the reduced amount of time given at each iteration. Therefore, one line of future research will deal with finding ways to overcome this problem, possibly by designing a self-adaptive version of the proposed algorithm. Note that parameter tuning might not be necessary anymore for such an algorithm version. In addition, introducing a learning mechanism that enables the CPLEX solutions to influence the probabilistic solution generation procedure is also considered as an important future research direction.

Acknowledgements

This work was funded by project CI-SUSTAIN, Spanish Ministry of Science and Innovation (PID2019-104156GB-I00). The corresponding author was funded by the Ministry of National Education, Turkey (Scholarship program: YLYS-2019).

References

[1] Christian Blum, Pedro Pinacho, Manuel López-Ibáñez, and José A Lozano. Construct, merge, solve & adapt: a new general algorithm for combinatorial optimization. *Computers & Operations Research*, 68:75–88, 2016.

[2] Christian Blum and Andrea Roli. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys*, 35(3):268–308, 2003.

[3] Salim Bouamama and Christian Blum. An improved greedy heuristic for the minimum positive influence dominating set problem in social networks. *Algorithms*, 14(3):79, 2021.

[4] MAI Fei and CHEN Weidong. An improved algorithm for finding minimum positive influence dominating sets in social networks. *Journal of South China Normal University*, 48(3):59–63, 2016.

[5] Martina Fischetti and Matteo Fischetti. *Matheuristics*, pages 121–153. Springer International Publishing, 2018.

[6] Angela K. Fournier, Erin Hall, Patricia Ricke, and Brittany Storey. Alcohol and the social network: Online social networking sites and college students' perceived drinking norms. *Psychology of Popular Media Culture*, 2(2):86, 2013.

[7] Dilek Günneç, Subramanian Raghavan, and Rui Zhang. Least-cost influence maximization on social networks. *INFORMS Journal on Computing*, 32(2):289–302, 2020.

[8] Geng Lin, Jian Guan, and Huibin Feng. An ilp based memetic algorithm for finding minimum positive influence dominating sets in social networks. *Physica A: Statistical Mechanics and its Applications*, 500:199–209, 2018.

[9] Geng Lin, Jinyan Luo, Haiping Xu, and Meiqin Xu. A hybrid swarm intelligence-based algorithm for finding minimum positive influence dominating sets. In Yong Liu, Lipo Wang, Liang Zhao, and Zhengtao Yu, editors, *Proceedings of ICNC-FSKD 2019 – Advances in Natural Computation, Fuzzy Systems and Knowledge Discovery*, pages 506–511. Springer International Publishing, 2020.

[10] Cheng Long and Raymond Chi-Wing Wong. Minimizing seed set for viral marketing. In *2011 IEEE 11th International Conference on Data Mining*, pages 427–436. IEEE press, 2011.

[11] Manuel López-Ibánez et al. The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, 3:43 – 58, 2016.

[12] Jiehui Pan and Tian-Ming Bu. A fast greedy algorithm for finding minimum positive influence dominating sets in social networks. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 360–364. IEEE, 2019.

[13] David Pisinger and Stefan Ropke. *Large Neighborhood Search*, pages 99–127. Springer International Publishing, 2019.

[14] Amir Afrasiabi Rad and Morad Benyoucef. Towards detecting influential users in social networks. In *International Conference on E-Technologies*, pages 227–240. Springer, 2011.

[15] Hassan Raei, Nasser Yazdani, and Masoud Asadpour. A new algorithm for positive influence dominating set in social networks. In *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 253–257. IEEE, 2012.

[16] Feng Wang, Erika Camacho, and Kuai Xu. Positive influence dominating set in online social networks. In *International Conference on Combinatorial Optimization and Applications*, pages 313–321. Springer, 2009.

[17] Feng Wang, Hongwei Du, Erika Camacho, Kuai Xu, Wonjun Lee, Yan Shi, and Shan Shan. On positive influence dominating sets in social networks. *Theoretical Computer Science*, 412(3):265–269, 2011.

[18] Guangyuan Wang. *Domination problems in social networks*. PhD thesis, University of Southern Queensland, 2014.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210113

From Non-Clausal to Clausal MinSAT

Chu-Min LI, [a]Felip MANYÀ, [b] Joan Ramon SOLER, [b] and Amanda VIDAL [b]
[a] *MIS, Université de Picardie, France*
[b] *Artificial Intelligence Research Institute, IIIA-CSIC, Spain*

Abstract. We tackle the problem of solving MinSAT for multisets of propositional formulas that are not necessarily in clausal form. Our approach reduces non-clausal to clausal MinSAT, since this allows us to rely on the much developed clause-based MinSAT solvers. The main contribution of this paper is the definition of several transformations of multisets of propositional formulas into multisets of clauses so that the maximum number of unsatisfied clauses in both multisets is preserved.

Keywords. Minimum satisfiability, clausal form, cost-preserving transformation.

1. Introduction

SAT is the problem of deciding whether there exists an assignment that satisfies a given (multi)set of propositional formulas. On the other hand, MaxSAT and MinSAT are optimization versions of SAT whose goal is to find an assignment that minimizes or maximizes the number of unsatisfied formulas, respectively. These problems are significant because many practical questions can be solved by first encoding them as a SAT, MaxSAT or MinSAT problems, and then finding a solution by solving the resulting encoding with a SAT, MaxSAT or MinSAT solver. While SAT is used to solve decision problems, MaxSAT and MinSAT are used to solve optimization problems [2,5].

We can distinguish between clausal SAT, MaxSAT, and MinSAT and non-clausal SAT, MaxSAT, and MinSAT, respectively. In the clausal case, the input multiset only contains clauses (i.e., disjunctions of literals). In the non-clausal case, the input multiset contains propositional formulas that are not necessarily in clausal form.

Many combinatorial problems admit more natural and compact encodings when represented in non-clausal form. However, the fastest and most robust SAT/MaxSAT/MinSAT solvers require their input in clausal form. Thus, some kind of clausal form transformation is needed to solve them. In SAT, there are several algorithms that transform a multiset of arbitrary propositional formulas into a satisfiability equivalent multiset of clauses [7,9]. Unfortunately, usual clausal form transformations used in SAT are not valid neither in MaxSAT nor MinSAT. The reason is that they are not cost-preserving; i.e., they do not preserve the minimum/maximum number of unsatisfied formulas between the input and the transformed multiset. It is therefore important to analyze how the existing SAT clausal form transformations behave for MaxSAT and MinSAT, as well as to investigate new cost-preserving transformations.

A few approaches to solve the non-clausal MaxSAT problem have been reported in [1,3,4], including one based on clausal form transformations [3]. To the best of our knowledge, no clausal form transformations have been defined for MinSAT. The only existing approach to solve Non-clausal MinSAT is via its reduction to Non-clausal MaxSAT [8]. Thus, in this paper, we focus on cost-preserving transformations from non-clausal to clausal MinSAT. More specifically, we define four transformations that, in practice, might produce different computational behavior.

2. Preliminaries

Given a set of propositional variables $\{x_1,\ldots,x_n\}$, a *literal* ℓ is a variable x_i or its negation $\neg x_i$, and a *clause* is a disjunction of literals. A *weighted clause* is a pair (c,w), where c is a clause and w, its weight, is a positive integer or infinity. If its weight is infinity, it is called a *hard clause* (we omit infinity weights for simplicity); otherwise, it is a *soft clause*. A *Weighted Partial MinSAT* instance is a finite multiset of weighted clauses. We represent MinSAT instances using multisets instead of sets because repeated clauses cannot be collapsed into one of such clauses as in SAT. Considering sets might affect the preservation of the minimum number of unsatisfied clauses.

Let the binary operations $\wedge,\vee$ be defined between (multi) sets of formulas as usual, namely for two multisets A,B and operation $\star$ either $\vee$ or $\wedge$, we let $A\star B := \{a\star b\colon a\in A, b\in B\}$. This naturally implies that the operations are associative and distributive over multisets. Moreover, for a multiset A and a single formula φ, we will use the convention that $A\star\varphi := A\star\{\varphi\}$. Lastly, $\cup$ or $\bigcup$ will denote the union of sets while $\sqcup$ and $\bigsqcup$ will denote the union of multisets, so that $\{a\}\cup\{a\}=\{a\}$ and $\{a\}\sqcup\{a\}=\{a,a\}$.

A *truth assignment* or evaluation is a mapping from the variables into $\{0,1\}$. We say it satisfies literal x $(\neg x)$ if x evaluates to 1 (0), weighted clause (c,w) if it satisfies a literal of c, and a multiset of clauses if it satisfies all its clauses. The weight w is interpreted as the penalty of violating clause c. When all clauses have the same weight, their weights will be omitted.

The Weighted Partial MaxSAT problem, or WPMaxSAT, for an instance ϕ, consists in finding an assignment that satisfies the hard clauses and minimizes the sum of the weights of the unsatisfied soft clauses. The Weighted Partial MinSAT problem, or WPMinSAT, is to find an assignment that satisfies the hard clauses and maximizes the sum of the weights of the unsatisfied soft clauses. The most common subproblems of WPMaxSAT are the following: Weighted MaxSAT (WMaxSAT), which is WPMaxSAT without hard clauses; Partial MaxSAT (PMaxSAT), which is WPMaxSAT when all the soft clauses have the same weight, and MaxSAT, which is PMaxSAT without hard clauses. Similarly, WMinSAT is WPMinSAT without hard clauses; PMinSAT is WPMinSAT when all the soft clauses have the same weight, and MinSAT is PMinSAT without hard clauses.

On the other hand, arbitrary propositional formulas are built in the usual way from a set of variables by using the binary connectives $\wedge,\vee$ and the unary connective $\neg$. Implication and bi-implication are binary connectives defined from the previous ones, by letting $\varphi\rightarrow\psi := \neg\varphi\vee\psi$ and $\varphi\leftrightarrow\psi := (\varphi\rightarrow\psi)\wedge(\psi\rightarrow\varphi)$. Propositional formulas are evaluated by considering the standard semantics of the connectives. When WPMaxSAT (WPMinSAT), as well as its simpler cases, take as input a multiset of propositional formulas, we refer to this problem as Non-clausal WPMaxSAT (WPMinSAT).

For what concerns this work, given a multiset of formulas A, $MinSAT(A)$ will formally denote the maximum number of unsatisfiable formulas in A.

Any propositional formula can be translated into Conjunctive Normal Form (CNF) through the following rules, which preserve logical equivalence: double negation elimination, De Morgan's laws and distributivity of $\vee$ over $\wedge$. Further simplifications[1] are also applied. This produces a formula in CNF, namely, a conjunction of clauses. In the sequel, $CNF(\varphi)$ denotes the formula equivalent to φ in CNF resulting from the previous equivalences, so naturally, for any truth-assignment e it holds that $e(\varphi) = e(CNF(\varphi))$.

When inquiring about the satisfiability of a set of propositional formulas $A = \{\varphi_1, \dots, \varphi_n\}$, each of the $CNF(\varphi_i)$ formulas can be split into a set of clauses (simply removing the conjunctions) to get a set of clauses that will serve as input to a SAT solver. We will refer to the union of these sets by $CNF_{SAT}(A)$. The satisfiability of $CNF_{SAT}(A)$ coincides with that of A, i.e., $SAT(A)$ if and only if $SAT(CNF_{SAT}(A))$. As we show below, this approach does not serve to solve non-clausal MinSAT or MaxSAT.

Example 2.1. *Let $A = \{(\neg x \leftrightarrow x) \wedge (\neg y \leftrightarrow y), x \vee y\}$ be a multiset of propositional formulas. Applying the transformation defined above, we get $CNF_{SAT}(A) = \{x, \neg x, y, \neg y, x \vee y\}$. The evaluation $e(x) = e(y) = 0$ violates two formulas of A and three clauses of $CNF_{SAT}(A)$, despite being an optimal MinSAT solution for both A and $CNF_{SAT}(A)$. Thus, that transformation is not cost preserving and $MinSAT(A) \neq MinSAT(CNF_{SAT}(A))$. Similarly, the evaluation $e(x) = e(y) = 1$ is an optimal MaxSAT solution for both A and $CNF_{SAT}(A)$, but violates one formula of A and two clauses of $CNF_{SAT}(A)$.*

Example 2.1 shows that the main problem arises when the CNF is split into clauses. This operation can generate additional clauses that violate the preservation of the maximum number of unsatisfied formulas. To overcome this drawback, we propose several new cost-preserving transformations for MinSAT in the next two sections.

3. Transformation CNF_{minSAT_d}

We first define a cost-preserving transformation for MinSAT, called CNF_{minSAT_d}, which does not add new variables. It has the advantage that it does not expand the solution space.

We define the partial mapping $^* : Fm \to MS$, where Fm stands for arbitrary formulas (but, as seen below, $*$ will be defined only over two particular families of formulas) and MS denotes the set of all multisets of clauses, as follows:

$$(c_1 \wedge c_2 \wedge \ldots \wedge c_n)^* := \{c_1, (\neg c_1)^* \vee c_2, \ldots, (\neg c_1)^* \vee (\neg c_2)^* \vee \ldots \vee (\neg c_{n-1})^* \vee c_n\}$$
$$(\neg(\ell_1 \vee \ell_2 \vee \ldots \ell_n))^* := \{\neg \ell_1, \ell_1 \vee \neg \ell_2, \ldots, \ell_1 \vee \ell_2 \vee \ldots \vee \ell_{n-1} \vee \neg \ell_n\}$$

for c_i denoting clauses and ℓ_i denoting literals.

Definition 3.1 (Transformation CNF_{minSAT_d}). *Let $A = A_{NC} \sqcup A_C$ be a multiset of formulas of which A_{NC} are not clauses and A_C are clauses. $CNF_{minSAT_d}(A)$ is the multiset of clauses*

$$\bigsqcup_{\varphi \in A_{NC}} CNF(\varphi)^* \sqcup A_C.$$

[1] e.g., removing clauses with a literal and its negation, or repeated clauses.

Example 3.2. *Given the multiset $A = \{\neg(\neg x_1 \wedge \neg x_2) \wedge (x_3 \vee x_4)\}$ formed by a single formula $\varphi = \neg(\neg x_1 \wedge \neg x_2) \wedge (x_3 \vee x_4)$, we have that $CNF(\varphi) = (x_1 \vee x_2) \wedge (x_3 \vee x_4)$. We convert it to a cost-preserving multiset of clauses for MinSAT as follows:*

$$\begin{aligned} CNF_{minSAT_d}(A) = ((x_1 \vee x_2) \wedge (x_3 \vee x_4))^* = \{x_1 \vee x_2, (\neg(x_1 \vee x_2))^* \vee (x_3 \vee x_4)\} = \\ \{x_1 \vee x_2, \{\neg x_1, x_1 \vee \neg x_2\} \vee (x_3 \vee x_4)\} = \\ \{x_1 \vee x_2, \neg x_1 \vee x_3 \vee x_4, \neg x_2 \vee x_3 \vee x_4\}. \end{aligned}$$

To prove that transformation CNF_{minSAT_d} is cost-preserving for MinSAT, we first study how translation $*$ behaves. Since it is, in an intuitive way, working from the inner-most level to the outer-most, let us first show the most internal level, and later, how the full translation works.

Lemma 3.3. *For a clause $c = \ell_1 \vee \ldots \vee \ell_n$ and a truth assignment e, $e(c) = 1$ if and only if all clauses $d \in (\neg c)^*$ are evaluated to 1 except for one.*

Proof. Since $e(c) = 1$ there is a minimum index $1 \leq i_0 \leq n$ for which $e(\ell_{i_0}) = 1$. For any $j < i_0$, since $e(\ell_j) = 0$, we have that $e(\ell_1 \vee \ell_2 \vee \ldots \vee \neg \ell_j) \geq e(\neg \ell_j) = 1$. Similarly, for any $j > i_0$, $e(\ell_1 \vee \ell_2 \vee \ldots \vee \neg \ell_j) \geq e(\ell_{i_0}) = 1$. □

Lemma 3.4. *For any formula ϕ and any truth-assignment e, it holds that $e(CNF(\phi)) = 0$ if and only if there is exactly one clause c in $CNF(\phi)^*$ such that $e(c) = 0$. Moreover, for any e, either e satisfies all clauses from $CNF(\phi)^*$ or it satisfies all but one clause.*

Proof. Let $CNF(\phi) = c_1 \wedge \ldots \wedge c_n$. First, assume that $e(CNF(\phi)) = 0$, and we will see there is exactly one falsified clause in $CNF(\phi)^*$. $e(c_1 \wedge \ldots \wedge c_n) = 0$ implies $i_0 = min\{i \in \{1, \ldots, n\} \colon e(c_i) = 0\}$ exists. Then, clearly for any $j < i_0$ and any $c \in (\neg c_1)^* \vee ... \vee (\neg c_{j-1})^* \vee c_j$, it holds that $e(c) = 1$, since by definition of disjuntion of multisets, any such c is of the form $\ldots \vee c_j$.

On the other hand, for $j > i_0$, any clause $c \in (\neg c_1)^* \vee ... \vee (\neg c_{j-1})^* \vee c_j$ has a subclause that belongs to $(\neg c_{i_0})^*$. We prove now that for any such $d \in (\neg c_{i_0})^*$ it also holds that $e(d) = 1$, thus implying $e(c) = 1$ too. Indeed, $c_{i_0} = l_1 \vee \ldots \vee l_s$ for some s, so $e(c_{i_0}) = 0$ implies that $e(l_i) = 0$ for all $1 \leq i \leq s$. Since every clause d in $\neg(c_{i_0})^*$ is by definition of the form $\ldots \vee \neg l_j$ for $1 \leq j \leq s$, necessarily $e(d) \geq e(\neg l_j) = 1$.

Now, from Lemma 3.3, we know that all clauses from $(\neg c_1)^* \vee ... \vee (\neg c_{i_0-1})^* \vee c_{i_0}$ are satisfied by e except for $d_1 \vee \ldots \vee d_{i_0-1} \vee c_{i_0}$ from $CNF(\phi)^*$, which is falsified.

On the other hand, to check that if one clause from $CNF(\phi)^*$ is falsified under e then also $CNF(\phi)$ is falsified, we can reason by contraposition. Assume $e(CNF(\phi)) = 1$, so $e(c_i) = 1$ for each $1 \leq i \leq n$. Since any clause d from $CNF(\phi)^*$ is of the form $\ldots \vee c_j$ for some $1 \leq j \leq n$, it is immediate that all clauses from $CNF(\phi)^*$ are satisfied.

Now, since any evaluation e can either satisfy or falsify $CNF(\phi)$, from the cases above we know e can either satisfy all clauses in $CNF(\phi)^*$, or satisfy all but one clause from $CNF(\phi)^*$. This concludes the proof. □

The previous lemma directly implies that, for any set of formulas A and truth assignment e, it holds that $|\{\phi \in A \colon e(\phi) = 0\}| = |\{\phi \in CNF_{minSAT_d}(A) \colon e(\phi) = 0\}|$.

Corollary 3.5. *$MinSAT(A) = MinSAT(CNF_{minSAT_d}(A))$.*

Proof. To check that $MinSAT(A) \leq MinSAT(CNF_{minSAT_d}(A))$, suppose that $MinSAT(A) = n$. Then, there is some truth assignment e such that $|\{\phi \in A\colon e(\phi) = 0\}| = n$. Thus, also $|\{\phi \in CNF_{minSAT_d}(A)\colon e(\phi) = 0\}| = n$, so the maximum number of unsatisfied formulas in $CNF_{minSAT_d}(A)$ is necessarily greater or equal than n.

The analogous reasoning in the other direction proofs $MinSAT(CNF_{minSAT_d}(A)) \leq MinSAT(A)$. □

Example 3.6. *Transformation CNF_{minSAT_d}, applied to the multiset of formulas $A = \{\neg(\neg x_1 \wedge \neg x_2) \wedge (x_3 \vee x_4)\}$ from Example 3.2, derived the multiset of clauses $\Phi = \{x_1 \vee x_2, \neg x_1 \vee x_3 \vee x_4, x_1 \vee \neg x_2 \vee x_3 \vee x_4\}$. Corollary 3.5 guarantees that the maximum number of unsatisfied formulas in A and Φ is preserved. In fact, for every evaluation e, the number of formulas unsatisfied by e in A and Φ is the same.*

4. Extending the Language and Relying on Partial MinSAT

We now define three cost-preservation transformations for MinSAT that add fresh variables to the language and rely on the partial MinSAT formalism to express some hard equivalences between the new variables and the non-clausal formulas of the input multiset A. We add the new variables y_ψ for some (sub)formulas ψ appearing in A or in $CNF_{SAT}(A)$, which will be specified for each translation. The difference among the three transformations lies in the formulas that receive a new variable and the way we encode the hard and soft constraints. For a given non-clausal MinSAT problem, the proposed transformations can generate multisets of clauses whose size ranges from polynomial to exponential in the length of the input formula. Moreover, the number of fresh variables can also be substantially different.

In what follows, we will use the following convention. For a set of formulas A and an arbitrary truth assignment e, we will let e' to be the modified truth assignment defined by letting $e'(p) = e(p)$ for all variables p in A, and $e'(y_\psi) = e(\psi)$ in all other cases.

4.1. Transformation CNF_{minSAT_e}

Definition 4.1 (Transformation CNF_{minSAT_e}). *Let $A = A_{NC} \sqcup A_C$ be a multiset of formulas of which A_{NC} are not a clauses and A_C are clauses. Let $\{y_\varphi\colon \varphi \in A_{NC}\}$ be a set of fresh variables not appearing in A. We call $CNF_{minSAT_e}(A)$ to the partial MinSAT instance given by*

$$\textbf{\textit{Hard clauses (HC)}} := \bigcup_{\varphi \in A_{NC}} CNF_{SAT}(\neg\varphi \vee y_\varphi), \quad \textbf{\textit{Soft clauses (SC)}} := \bigsqcup_{\varphi \in A_{NC}} \{y_\varphi\} \cup A_C$$

Notice that the formulas $\varphi \in A_{NC}$ occur negated in $CNF_{SAT}(\neg\varphi \vee y_\varphi)$. This is relevant because the formulas φ occur with positive polarity in transformation CNF_{minSAT_d}. This implies that we could avoid the combinatorial explosion due to the application of distributivity of $\vee$ over $\wedge$ if the most appropriate transformation is chosen for each formula.

Example 4.2. *For the multiset $A = \{\neg(\neg x_1 \wedge \neg x_2) \wedge (x_3 \vee x_4)\}$ of Example 3.2, we have that $\neg\varphi \vee y_\varphi = \neg(\neg(\neg x_1 \wedge \neg x_2) \wedge (x_3 \vee x_4)) \vee y_\varphi$. The application of Transformation CNF_{minSAT_e} derives the following partial MinSAT instance:*

Hard clauses: $\neg x_1 \vee \neg x_3 \vee y_\varphi \quad \neg x_1 \vee \neg x_4 \vee y_\varphi \quad \neg x_2 \vee \neg x_3 \vee y_\varphi \quad \neg x_2 \vee \neg x_4 \vee y_\varphi$
Soft clauses: y_φ

In the proofs below for checking that the previous translation is cost preserving, with the objective of lightening the notation, we will assume A_C is empty, since it is clear it does not affect the calculations.

Lemma 4.3. *Let A be a multiset of formulas and let e be a truth assignment such that $e(c) = 1$ for all $c \in \mathbf{HC}(CNF_{minSAT_e}(A))$. Then,*

$$|\{c \in \mathbf{SC}(CNF_{minSAT_e}(A))\colon e(c) = 0\}| \leq |\{\varphi \in A\colon e(\varphi) = 0\}|.$$

Proof. By assumption $e(c) = 1$ for all $c \in CNF_{SAT}(\neg\varphi \vee y_\varphi)$ for each $\varphi \in A$. Since for an arbitrary formula ψ the conjunction of all clauses $CNF_{SAT}(\psi)$ equals ψ, it is clear that $e(\neg\varphi \vee y_\varphi) = 1$ for all $\varphi \in A$, implying that $e(\varphi) \leq e(y_\varphi)$. Then, for each $\varphi \in A$ such that $e(y_\varphi) = 0$, necessarily $e(\varphi) = 0$. □

On the other hand, for any set of formulas A and evaluation e, we can prove that the evaluation e' defined at the start of Section 4 is such that the number of falsified formulas of A under e and that of falsified soft clauses in $CNF_{minSAT_e}(A)$ under e' is the same.

Lemma 4.4. *Let A be a multiset of formulas and let e be an arbitrary truth assignment. Then, e' satisfies all hard clauses in $CNF_{minSAT_e}(A)$ and*

$$|\{\phi \in A\colon e(\phi) = 0\}| = |\{c \in \mathbf{SC}(CNF_{minSAT_e}(A))\colon e'(c) = 0\}|.$$

Proof. Observe that the truth assignment e' is defined in such a way that it preserves evaluation for all variables in A (and so, for all formulas in A), and where $e'(y_\varphi) = e(\varphi)$. Then, for any $\varphi \in A$ we have that $e'(\neg\varphi \vee y_\varphi) = e'(\neg\varphi) \vee e'(y_\varphi) = e(\neg\varphi) \vee e(\varphi) = 1$. Thus, all hard clauses in $CNF_{minSAT_e}(A)$ hold.

For what concerns the second statement of the lemma, we only need to check that $|\{\phi \in A\colon e(\phi) = 0\}| = |\{y_\phi \in \mathbf{SC}(CNF_{minSAT_e}(A))\colon e'(y_\phi) = 0\}|$. This is immediate, since $e'(y_\phi) = e(\phi)$ by definition. □

Corollary 4.5. $MinSAT(A) = MinSAT(CNF_{minSAT_e}(A))$.

Proof. To prove $\leq$ observe that $MinSAT(A) = n$ implies there is some truth assignment e for which $|\{\phi \in A\colon e(\phi) = 0\}| = n$. Then, Lemma 4.4 implies that $|\{c \in \mathbf{SC}(CNF_{minSAT_e}(A))\colon e'(c) = 0\}| = n$ too. Since $MinSAT(CNF_{minSAT_e}(A))$ is the maximum number of falsifiable clauses, we get that, in particular, $MinSAT(CNF_{minSAT_e}(A)) \geq n$ too, so $MinSAT(A) \leq MinSAT(CNF_{minSAT_e}(A))$.

Symmetrically, to prove $\geq$ observe that, by definition, $MinSAT(CNF_{minSAT_e}(A)) = |\{c \in \mathbf{SC}(CNF_{minSAT_e}(A))\colon g(c) = 0\}|$ for a certain truth assignment g that moreover satisfies all hard clauses in $CNF_{minSAT_e}(A)$. Applying Lemma 4.3 we know that, for such g, it holds that $|\{c \in \mathbf{SC}(CNF_{minSAT_e}(A))\colon g(c) = 0\}| \leq |\{\varphi \in A\colon g(\varphi) = 0\}|$. Thus, $|\{\varphi \in A\colon g(\varphi) = 0\}| \geq n$. Again, since $MinSAT(A)$ is the maximum of falsifiable formulas from A, in particular $MinSAT(A) \geq n$ too, meaning that $MinSAT(A) \geq MinSAT(CNF_{minSAT_e}(A))$, concluding the proof. □

4.2. Transformation CNF_{minSAT_i}

Definition 4.6 (Transformation CNF_{minSAT_i}). *Let $A = A_{NC} \sqcup A_C$ be a multiset of formulas of which A_{NC} are not clauses and A_C are clauses. Let $\{y_c \colon \varphi \in A_{NC}, c \in CNF_{SAT}(\varphi)\}$ be a set of fresh variables not appearing in A.*

We call $CNF_{minSAT_i}(A)$ to the partial MinSAT instance given by

$$\textbf{\textit{Hard clauses (HC)}} := \bigcup_{\varphi \in A_{NC}} \bigcup_{c \in CNF_{SAT}(\varphi)} \{\neg c \vee y_c\}$$

$$\textbf{\textit{Soft clauses (SC)}} := \bigsqcup_{\varphi \in A_{NC}} (\bigwedge_{c \in CNF_{SAT}(\varphi)} y_c)^*$$

Example 4.7. *For the multiset $A = \{(\neg x_1 \to x_2) \wedge (x_1 \to \neg x_2)\}$, transformation $CNF_{minSAT_i}(A)$ derives the following partial MinSAT instance:*

$$\begin{array}{lllll} \textit{Hard clauses:} & \neg x_1 \vee y_{c_1} & \neg x_2 \vee y_{c_1} & x_1 \vee y_{c_2} & x_2 \vee y_{c_2} \\ \textit{Soft clauses:} & y_{c_1} & \neg y_{c_1} \vee y_{c_2} & & \end{array}$$

Observe that $CNF_{minSAT_i}(A)$ associates one fresh variable with every clause of $CNF_{SAT}(\varphi)$. If only one fresh variable is associated with each CNF, we would have as hard clauses $\neg x_1 \vee y_{c_1}, \neg x_2 \vee y_{c_1}, x_1 \vee y_{c_1}, x_2 \vee y_{c_1}$, and we would get an infeasible solution if y_{c_1} is unsatisfied because we would detect a contradiction in the hard part.

If we consider the multiset $A' = \{(\neg x_1 \to x_2), (x_1 \to \neg x_2)\}$ containing two formulas, transformation $CNF_{minSAT_i}(A)$ derives the following partial MinSAT instance:

$$\begin{array}{lllll} \textit{Hard clauses:} & \neg x_1 \vee y_{c_1} & \neg x_2 \vee y_{c_1} & x_1 \vee y_{c_2} & x_2 \vee y_{c_2} \\ \textit{Soft clauses:} & y_{c_1} & y_{c_2} & & \end{array}$$

Despite the similarity between A and A', the maximum number of unsatisfied clauses is 1 in A and 2 in A'. They have the same hard part, but the soft clauses are different.

Similarly to what we did in the previous subsection, we will prove the cost preservation assuming $A_C = \emptyset$. We prove an analogous version of Lemma 4.3, but resorting to Lemma 3.4 to keep track of the behavior of the new soft clauses.

Lemma 4.8. *Let A be a multiset of formulas and let e be a truth assignment such that $e(c) = 1$ for all $c \in \mathbf{HC}(CNF_{minSAT_i}(A))$. Then,*

$$|\{c \in \mathbf{SC}(CNF_{minSAT_i}(A)) \colon e(c) = 0\}| \leq |\{\varphi \in A \colon e(\varphi) = 0\}|.$$

Proof. By assumption $e(\neg c \vee y_c)$ for all $c \in CNF_{SAT}(\varphi)$ with $\varphi \in A$. Thus, $e(c) \leq e(y_c)$ for all such c and y_c. Moreover, observe that each formula of the form $\bigwedge_{c \in CNF_{SAT}(\varphi)} y_c$ is already in conjunctive normal form. Thus, by Lemma 3.4, $e(\bigwedge_{c \in CNF_{SAT}(\varphi)} y_c) = 0$ if and only if exactly one clause from $(\bigwedge_{c \in CNF_{SAT}(\varphi)} y_c)^*$ is falsified, and otherwise all clauses in $(\bigwedge_{c \in CNF_{SAT}(\varphi)} y_c)^*$ are satisfied by e.

By definition,

$$|\{c \in \mathbf{SC}(CNF_{minSAT_i}(A))\colon e(c) = 0\}| = \sum_{\varphi \in A} |\{d \in (\bigwedge_{c \in CNF_{SAT}(\varphi)} y_c)^*\colon e(d) = 0\}|.$$

The previous observation implies that $|\{d \in (\bigwedge_{c\in CNF_{SAT}(\varphi)} y_c)^*\colon e(d) = 0\}| \leq 1$. Moreover, $|\{d \in (\bigwedge_{c\in CNF_{SAT}(\varphi)} y_c)^*\colon e(d) = 0\}| = 1$ if and only if $e(\bigwedge_{c\in CNF_{SAT}(\varphi)} y_c) = 0$, and so, there is at least some $c \in CNF_{SAT}(\varphi)$ for which $e(y_c) = 0$ too. Since we saw above that $e(c) \leq e(y_c)$, we have that $e(c) = 0$, making $e(\varphi) = 0$ too. □

We will see in the next lemma the analogous to Lemma 4.4. Indeed, we can easily check that the number of unsatisfied formulas of A under e coincides with the number of unsatisfied soft clauses in $CNF_{minSAT_i}(A)$ under e' (for e' as defined in the beginning of Section 4). It will only be necessary to rely on Lemma 3.4 when necessary.

Lemma 4.9. *Let A be a multiset of formulas and let e be a truth assignment. Then, e' satisfies all hard clauses in $CNF_{minSAT_i}(A)$ and $|\{\phi \in A\colon e(\phi) = 0\}| = |\{c \in \mathbf{SC}(CNF_{minSAT_i}(A))\colon e'(c) = 0\}|$.*

Proof. Observe that the truth assignment e' is defined in such a way that it preserves evaluation for all variables in A (and so, for all formulas in A), and where $e'(y_c) = e(c)$ for all $c \in CNF_{SAT}(\varphi)$ for $\varphi \in A$. Then, for any $\varphi \in A$ we have that $e'(\neg c \vee y_\varphi) = e'(\neg c) \vee e'(y_c) = e(\neg c) \vee e(c) = 1$. Thus, all hard clauses in $CNF_{minSAT_i}(A)$ hold.

For what concerns the second statement of the lemma, we only need to check that

$$|\{\phi \in A\colon e(\phi) = 0\}| = |\{d \in (\bigwedge_{c \in CNF_{SAT}(\phi)} y_c)^*\colon \phi \in A, e'(d) = 0\}|.$$

By Lemma 3.4, for each $\phi \in A$ there are two cases:

- $|\{d \in (\bigwedge_{c\in CNF_{SAT}(\phi)} y_c)^*\colon e'(d) = 0\}| = 1$, which happens if and only if $e'(\bigwedge_{c\in CNF_{SAT}(\phi)} y_c) = 0$, and so, since $e'(y_c) = e(c)$ for each such c, if and only if $e(\phi) = 0$ too, or
- $|\{d \in (\bigwedge_{c\in CNF_{SAT}(\phi)} y_c)^*\colon e'(d) = 0\}| = 0$, which implies, following the same reasoning, that $e(\phi) = 1$. □

Corollary 4.10. *$MinSAT(A) = MinSAT(CNF_{minSAT_e}(A))$*

Proof. To prove $\leq$ observe that $MinSAT(A) = n$ implies there is some truth assignment e for which $|\{\phi \in A\colon e(\phi) = 0\}| = n$. Then Lemma 4.4 implies that $|\{c \in \mathbf{SC}(CNF_{minSAT_e}(A))\colon e'(c) = 0\}| = n$ too. Since $MinSAT(CNF_{minSAT_e}(A))$ is the maximum number of falsifiable clauses, we get that, in particular, $MinSAT(CNF_{minSAT_e}(A)) \geq n$ too, so $MinSAT(A) \leq MinSAT(CNF_{minSAT_e}(A))$.

Symmetrically, to prove $\geq$ observe that, by definition, $MinSAT(CNF_{minSAT_e}(A)) = |\{c \in \mathbf{SC}(CNF_{minSAT_e}(A))\colon e(c) = 0\}|$ for a certain truth assignment e that moreover satisfies all hard clauses in $CNF_{minSAT_e}(A)$. Applying Lemma 4.3 we know that, for such g, it holds that $|\{c \in \mathbf{SC}(CNF_{minSAT_e}(A))\colon g(c) = 0\}| \leq |\{\varphi \in A\colon g(\varphi) = 0\}|$. Thus, $|\{\varphi \in A\colon g(\varphi) = 0\}| \geq n$. Again, since $MinSAT(A)$ is the maximum of falsifiable formulas from A, in particular $MinSAT(A) \geq n$ too, meaning that $MinSAT(A) \geq MinSAT(CNF_{minSAT_e}(A))$, concluding the proof. □

4.3. Transformation CNF_{minSAT_t}

To avoid the generation of a number of clauses that can be exponential in the number of input formulas due to the application of distributivity when deriving the CNF, we can adapt the Tseitin transformation to MinSAT. We begin by recalling the Tseitin transformation for SAT, and then point out how it is adapted to MinSAT.

For each formula φ, let us denote by $SFm(\varphi)$ the set of subformulas of φ. Then, for each $\psi \in SFm(\varphi)$ consider a new variable y_ψ, and for each such formula ψ we define the set of clauses $Def(\psi)$ by[2]

$$\begin{aligned} Def(p) &:= \emptyset && \text{for } p \text{ propositonal variable,} \\ Def(\psi \star \chi) &:= CNF_{SAT}(y_{\psi\star\chi} \leftrightarrow y_\psi \star y_\chi) && \text{for } \star \in \{\vee, \wedge\}, \\ Def(\neg\psi) &:= CNF_{SAT}(y_{\neg\psi} \leftrightarrow \neg y_\psi) \end{aligned}$$

Recall that all connectives different from $\wedge, \vee, \neg$ are simply wrapping some expression involving these three, so any formula φ is written in fact in the previous language and so, Def is correctly defined for all formulas. It is clear that the above definitions generate clauses with at most 3 literals each. Then, for a formula φ, its Tseitin SAT transformation $T(\varphi)$ is the set of clauses $T(\varphi) := \{y_\varphi\} \cup \bigcup_{\psi \in SFm(\varphi)} Def(\psi)$.

It is rather standard to check that $SAT(\varphi)$ if and only if $SAT(T(\varphi))$, and thus, also for a set of formulas A, we have that $SAT(A)$ if and only if $SAT(T(A))$, where, as usual, by $T(A)$ we denote the set $\bigcup_{\varphi \in A} T(\varphi)$.

To use this transformation in order to preserve $MinSAT$, it is only necessary to do a slight modification to the previous transformation and rely on partial $MinSAT$. Let us consider the transformation $T^-(\varphi)$ given by

$$T^-(\varphi) := \bigcup_{\psi \in SFm(\varphi)} Def(\psi)$$

Observe this is simply removing from the resulting set of clauses the outermost variable, which is the one that, at SAT, is imposing that the corresponding formula is satisfied. Similarly to CNF_{minSAT_e}, we define the transformation CNF_{minSAT_t}.

Definition 4.11 (Transformation CNF_{minSAT_t})**.** *Let $A = A_{NC} \sqcup A_C$ be a multiset of formulas of which A_{NC} are not clauses and A_C are clauses. We call $CNF_{minSAT_t}(A)$ to the partial MinSAT instance given by*

$$\textbf{\textit{Hard clauses (HC)}} := \bigcup_{\varphi \in A_{NC}} T^-(\varphi), \qquad \textbf{\textit{Soft clauses (SC)}} := \bigsqcup_{\varphi \in A_{NC}} \{y_\varphi\} \sqcup A_C$$

Example 4.12. *Given the multiset of formulas $A = \{x_1 \wedge x_2, x_3 \wedge x_4\}$, $CNF_{minSAT_t}(A)$ derives the following partial MinSAT instance:*

Hard clauses: $\neg y_1 \vee x_1 \quad \neg y_1 \vee x_2 \quad y_1 \vee \neg x_1 \vee \neg x_2 \quad \neg y_2 \vee x_3 \quad \neg y_2 \vee x_4 \quad y_2 \vee \neg x_3 \vee \neg x_4$
Soft clauses: $y_1 \quad y_2$

[2] To simplify the notation, we do not distinguish between propositional variables and more complex formulas. Thus, a variable p will receive a fresh variable y_p in the new language, and p will no longer appear in the translation. The definition of Def over propositional variables is also included to lighten notation later on.

Lemma 4.13. $MinSAT(A) = MinSAT(CNF_{minSAT_t}(A))$

Proof. To prove $\leq$ observe that $MinSAT(A) = n$ implies there is some truth assignment e for which $|\{\phi \in A\colon e(\phi) = 0\}| = n$. As we did in Lemma 4.4 for CNF_{minSAT_e}, we can easily check that $|\{c \in \mathbf{SC}(CNF_{minSAT_t}(A))\colon e'(c) = 0\}| = n$ too. Indeed, since $e'(y_\psi) = e(\psi)$ for all subformula ψ of formulas in A, clearly all hard clauses are satisfied under e'. Moreover, since the multiset of soft clauses is formed exactly by the singletons y_ϕ for each $\phi \in A$ (as a multiset), then necessarily $|\{c \in \mathbf{SC}(CNF_{minSAT_t}(A))\colon e'(c) = 0\}| = n \leq MinSAT(CNF_{minSAT_t}(A))$, since MinSAT is the maximum number of falsifiable formulas. Thus, $MinSAT(A) \leq MinSAT(CNF_{minSAT_t}(A))$.

On the other hand, similarly again to the case for CNF_{minSAT_e}, to prove $\geq$ observe that, by definition, $MinSAT(CNF_{minSAT_t}(A)) = |\{c \in \mathbf{SC}(CNF_{minSAT_t}(A))\colon g(c) = 0\}|$ for a certain truth assignment g that moreover satisfies all hard clauses in $CNF_{minSAT_t}(A)$. Let us modify g to cope with the original variables in A, simply by letting $g'(p) = g(y_p)$ for all propositional variables p appearing in A. Then, as it happens in the usual Tseitin transformation, is immediate that $g'(\phi) = g(y_\phi)$ for all $\phi \in A$. Thus, we have that $|\{c \in \mathbf{SC}(CNF_{minSAT_t}(A))\colon g(c) = 0\}| = |\{\phi \in A\colon g'(\phi) = 0\}| \leq MinSAT(A)$. □

5. Conclusions and Future Work

We proposed the first approach to solve non-clausal MinSAT via its reduction to clausal MinSAT. We defined four clausal transformations (CNF_{minSAT_d}, CNF_{minSAT_e}, CNF_{minSAT_i} and CNF_{minSAT_t}) and proved its correctness. The most immediate future work is to conduct an empirical comparison and extend the results to finite-domain variables [6].

Acknowledgements: Research partially supported by the Spanish State Research Agency (AEI) project PID2019-111544GB-C2, and CSIC's i-Link project LINKC20018.

References

[1] G. Fiorino. New tableau characterizations for non-clausal MaxSAT problem. *Logic Journal of the IGPL*, 2021.

[2] C. M. Li and F. Manyà. MaxSAT, hard and soft constraints. In A. Biere, H. van Maaren, and T. Walsh, editors, *Handbook of Satisfiability*, pages 613–631. IOS Press, 2009.

[3] C. M. Li, F. Manyà, and J. R. Soler. Clausal form transformation in MaxSAT. In *Proceedings of the 49th IEEE International Symposium on Multiple-Valued Logic, ISMVL*, pages 132–137, 2019.

[4] C. M. Li, F. Manyà, and J. R. Soler. A tableau calculus for non-clausal maximum satisfiability. In *Proceedings of the 28th International Conference on Automated Reasoning with Analytic Tableaux and Related Methods, TABLEAUX*, pages 58–73, 2019.

[5] C. M. Li, Z. Zhu, F. Manyà, and L. Simon. Optimizing with minimum satisfiability. *Artificial Intelligence*, 190:32–44, 2012.

[6] F. Manyà. The 2-SAT problem in signed CNF formulas. *Multiple-Valued Logic. An International Journal*, 5(4):307–325, 2000.

[7] D. A. Plaisted and S. Greenbaum. A structure-preserving clause form translation. *Journal of Symbolic Computation*, 2:293–304, 1986.

[8] J. R. Soler. *New Solving Techniques for Maximum and Minimum Satisfiability*. PhD thesis, UAB, 2021.

[9] G. Tseitin. *Studies in Constructive Mathematics and Mathematical Logic, Part II*, chapter On the Complexity of Derivations in the Propositional Calculus, pages 115–125. Steklov Mathematical Inst., 1968.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210114

Perceptual Maps to Aggregate Information from Decision Makers

Olga PORRO [a], Francesc PARDO-BOSCH [a], Mónica SÁNCHEZ [a,1] and Núria AGELL [b]
[a] *UPC-BarcelonaTech, Barcelona, Spain*
[b] *Esade - URL, Barcelona, Spain*

Abstract. Understanding different perceptions of human being when using linguistic terms is a crucial issue in human-machine interaction. In this paper, we propose the concept of perceptual maps to model human opinions in a group decision-making context. The proposed approach considers a multi-granular structure using unbalanced hesitant linguistic term sets. An illustrative case is presented in the location decisions made by multinationals enterprises of the energy sector within the European smart city context.

Keywords. Fuzzy systems, hesitant linguistic terms, multi-criteria decision-aiding.

1. Introduction

A better interaction between humans and intelligent systems needs being able to capture some human abilities such as asking questions or constructing explanations. Humans consider real world situations from different perspectives. And considering this heterogeneity could revert in better group decision making.

Multiple-criteria group decision making (MCGDM) is used when a group of experts or decision makers (DMs) express their assessments or preferences on a set of attributes (or criteria) for a set of alternatives and an optimal representative or solution is needed to solve the problem [3]. Many practical applications have used hesitant fuzzy linguistic term sets (HFLTSs) to deal with the linguistic information involved in MCGDM problems. Most of the GDM applications found in the literature, which are framed as MCDM problems with linguistic assessments modelled by means of HFLTSs, are assumed to be built over a uniform and symmetrically distributed linguistic term set (LTS). However, there exist many GDM situations where attributes relate to qualitative characteristics that need to be assessed by linguistic terms represented by unsymmetrical or not uniformly distributed LTSs, i.e., unbalanced LTS, such as for example, the evaluation of creditworthiness and credit risk quality of bonds [2] or factors affecting the comfort of passengers [4]. Similarly, it is also very common to find GDM situations

[1] Corresponding Author: Mónica Sánchez, UPC-BarceonaTech, Barcelona, Spain; E-mail: monica.sanchez@upc.edu

with DMs having different backgrounds or knowledge and this also needs to be modelled by different LTS. In this paper, these differences between the DMs' semantics of the linguistic term set are represented via the concept of perceptual maps. This concept, together with the idea of the gap measurement in the proposed distance, represent a step forward with respect to the state-of-the-art due to its flexibility to model not only hesitancy but also discrepancies between DMs' assessments.

The main goal of this paper is to show the feasibility and practicability of a fuzzy decision-aiding approach, using multi-granular and unbalanced hesitant fuzzy linguistic term sets in a real world multi criteria group decision making situation involving several experts who elicitate their opinions with linguistic assessments. An illustrative application is presented in this paper, framed in the scheme of location decisions made by multinationals enterprises (MNEs) of the energy sector within the European Smart city context.

The rest of the paper is organized as follows. Section 2 introduces preliminary concepts. Section 3 presents the illustrative case and the decision-aiding approach considering perceptual maps to aggregate information from decision makers. Finally, the conclusions and future work are presented in Section 4.

2. Preliminaries

Hesitant fuzzy linguistic term sets were introduced in [10]. They are useful to capture the human way of reasoning using linguistic expressions involving different levels of precision. Based on this concept, in this section, a formal introduction to perceptual maps and projected space to aggregate decision makers' opinions is presented.

Let S be a finite totally ordered set of linguistic terms, $\mathcal{S} = \{s_1, \ldots, s_n\}$, with $s_1 < \ldots < s_n$.

Definition 1 *([10])* A *hesitant fuzzy linguistic term set (HFLTS)* over $\mathcal{S}$ is a subset of consecutive linguistic terms of S, i.e. $\{x \in \mathcal{S} \,|\, a_i \leq x \leq a_j\}$, for some $i, j \in \{1, \ldots, n\}$ with $i \leq j$. We also consider the *empty HFLTS:* $\{\} = \emptyset$*, and the full HFLTS:* $\mathcal{S}$*.*

Hereafter, the non-empty HFLTS $H = \{x \in \mathcal{S} \,|\, a_i \leq x \leq a_j\} = \{a_i, a_{i+1}, \ldots, a_j\}$ is also denoted by $[a_i, a_j]$. If $i = j$, $[a_i, a_i]$ is the singleton $\{a_i\}$. The set of all HFLTSs over S is denoted by $\mathbb{S}_n$: $\mathbb{S}_n = \{[a_i, a_j] \mid i, j \in \{1, \ldots, n\}, i \leq j\} \cup \{\emptyset\}$, being $\mathbb{S}_n^* = \mathbb{S}_n - \{\emptyset\}$ the set of non-empty HFLTSs. The binary operation, *connected union,* $\sqcup$, of two HFLTSs was defined in [6] as the least element of $\mathbb{S}_n$, based on the subset inclusion relation $\subseteq$, that contains both HFLTSs. The intersection, $\sqcap$, of HFLTSs is a closed binary operation on the set $\mathbb{S}_n$. In ([6]) it is proved that $(\mathbb{S}_n, \sqcup, \sqcap)$ is a non-distributive lattice.

Next, we consider the concept of a normalized measure over a linguistic term set S, which may not be balanced: the perceptual map.

2.1. Perceptual map

Definition 2 *[8]* Let S be a totally ordered finite LTS, $S = \{s_1, s_2, ..., s_n\}$. Let μ' denote a measure over S such that $\mu'(s_i) > 0$, $\forall i \in \{1, 2, \ldots, n\}$. Then, the *perceptual map*, μ, induced by μ', is a function $\mathbb{S}_n^* \to [0, 1]$ defined as:

$$\mu(H_S) = \frac{\sum\limits_{s_i \in H_S} \mu'(s_i)}{\sum\limits_{i=1}^{n} \mu'(s_i)} \tag{1}$$

for any $H_S \in \mathbb{S}_n^*$.

The perceptual map μ provides a normalized measure on $\mathbb{S}_n^*$ and, as highlighted in [9], there exists a bijective function between the set of perceptual maps over a set S of granularity n and the set of partitions of $[0, 1]$ with n non-empty subintervals. Indeed, given a partition of the interval $[0, 1]$ defined by the strictly ordered n-tuple of real numbers P=$\{\alpha_0, \cdots, \alpha_i, \cdots, \alpha_n\}$ such that $0 = \alpha_0$ and $\alpha_n = 1$, there exist a perceptual map μ defined over a LTS with granularity n, with $\mu(s_i) > 0$ and $\alpha_i - \alpha_{i-1} = \mu(s_i)$, $\forall i \in \{1, \cdots, n\}$.

Note that there are several methods, either supervised or non-supervised, to define the landmarks of the partition and consequently the perceptual map underlying different decision making styles.

An extended lattice of HFLTSs $\overline{\mathbb{S}_n} = \mathbb{S}_n^* \cup \mathcal{A} \cup (-\mathbb{S}_n^*)$ was defined in [6], considering the set $\mathcal{A}$ of null HFLTSs and the set $-\mathbb{S}_n^*$ of negative HFLTSs. Note that, intuitively, null HFLTSs correspond to landmarks and negative HFLTSs modelize the gap between a pair of disjoint HFLTSs. This extension allows us to consider a distance between HFLTSs that takes into account the gap between two HFLTSs if they do not overlap.

2.2. A perceptual-based distance for unbalanced HFLTSs

Given any perceptual map, μ, the definition of width of a HFLTS $H_S \in \overline{\mathbb{S}_n}$ and the distance between two HFLTSs H_S^1, $H_S^2 \in \overline{\mathbb{S}_n}$ are defined in [8] as follows:

$$W_\mu(H_S) = \begin{cases} \mu(H_S), & H_S \in \mathbb{S}_n^*; \\ 0, & H_S \in \mathcal{A}; \\ -\mu(-H_S), & H_S \in (-\mathbb{S}_n^*). \end{cases}$$

Let H_S^1, $H_S^2 \in \overline{\mathbb{S}_n}$, then:

$$D_\mu(H_S^1, H_S^2) = W_\mu(H_S^1 \sqcup H_S^2) - W_\mu(H_S^1 \sqcap H_S^2) \tag{2}$$

provides a distance in the lattice ($\overline{\mathbb{S}_n}$,$\sqcup$,$\sqcap$), were $\sqcup$ and $\sqcap$ are the extended union and extended intersection respectively [8]. Note that this distance considers the gap between two HFLTSs if they do not overlap.

2.3. A projected space for multi-perceptual GDM

In order to compare and operate with unbalanced HFLTSs based on different perceptual maps, a projected space is defined to project linguistic assessments built over different perceptual maps onto a projected linguistic structure [8], specifically:

Definition 3 *([8])* Let $\{P_{\mu_j} \mid j \in 1, \ldots, k\}$ be the set of partitions associated to the set of perceptual maps $\{\mu_j \mid j \in 1, \ldots, k\}$ and the set of LTS $\{S_{\mu_j} \mid j \in 1, \ldots, k\}$. Each P_{μ_j} is a partition of the unit interval defined by $\{\lambda_0^j, \lambda_1^j, \lambda_2^j, \cdots, \lambda_{n_j}^j\}$, with $\lambda_0^j = 0$, $\lambda_{n_j}^j = 1$ and n_j denotes the cardinality of each S_j. The *projected partition* associated to $\{P_{\mu_j} \mid j \in 1, 2, \ldots, k\}$ is P_p, defined by $\bigcup_{j=1}^{k} \bigcup_{l=0}^{n_j} \{\lambda_l^j\}$.

Definition 4 *([8])* Let P_p be the projected partition of the set $\{P_{\mu_j} \mid j \in 1, 2, \ldots, k\}$ defined by $\{\lambda_0, \lambda_1, \ldots, \lambda_{n^*}\}$. We define the *projected LTS*, S^*, as the set that contains the *projected basic labels*, s_α^*, i.e., $S^* = \{s_\alpha^* \mid \alpha \in 1, 2 \ldots, n^*\}$, where n^* is the cardinality of the set $\bigcup_{j=1}^{k} \bigcup_{l=0}^{n_j} \{\lambda_l^j\}$ *; and the projected normalized measure over* S^**,* μ'_* *induced by this partition as:*

$$\mu'_*(s_\alpha^*) = \lambda_\alpha - \lambda_{\alpha-1}, \alpha \in 1, 2, \ldots, n^* \tag{3}$$

where $s_\alpha^* \in S^*$.

Note that the projected basic labels are only considered for computational purposes and the semantics that apply to each S_j do not apply for S^*. From the above concepts, we lastly introduce the concept of projected perceptual map:

Definition 5 *([8])* Let S^* be a projected LTS, $S^* = \{s_1^*, s_2^*, ..., s_{n^*}^*\}$ and μ'_* its projected normalized measure. Then, the *projected perceptual map* μ_* is the perceptual map induced by μ'_* in $\mathcal{H}_{S^*}^*$ (as defined in equation 1).

Note that the previous definitions not only deal with unbalanced LTS but are also adapted to contexts of multi-granularity when the LTS used by each DM, S_j, are of different cardinality.

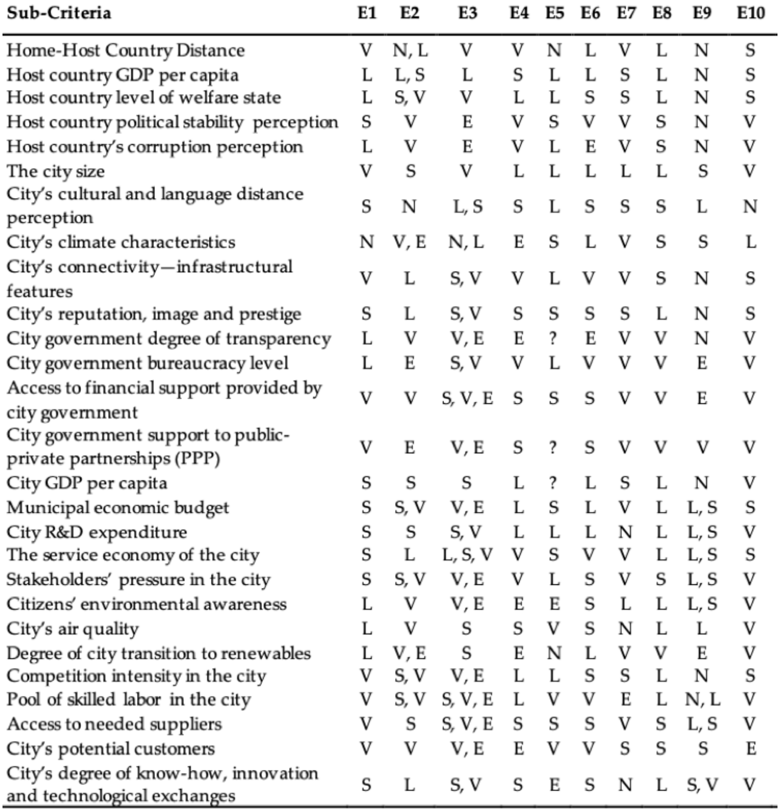
The previously described perceptual map and projected algebraic structure, will allow us to deal with MCGDM problems where each decision maker has its own qualitative reasoning approach. In this direction, an adaptation of the Fuzzy TOPSIS method was defined in [8,7] to rank different alternatives when considering a framework where DMs are allowed to use different perceptual maps.

3. Illustrative Case: Energy MNEs locations

Energy multinational enterprises (MNE) provide green energy services and products to the city they are located. In addition, as the rest of MNEs, they generate jobs and stimulate local economy. Understanding the multi-criteria decision making process followed by these energy enterprises in their location decisions would facilitate local governments to attract them.

In this paper, an extension of the study presented in [7] is conducted conseder-ing different perceptual maps. We use the concepts defined in the previous section to design a framework for ranking criteria and sub-criteria governing the energy MNEs strategic location decisions. The challenge of understanding the process that energy MNEs follow in their location decisions is studied using extended fuzzy multi-perceptual linguistic TOPSIS method.

The framework is build based on extracting the relative importance of 27 sub-criteria by asking the opinion to ten experts of the field. Subcriteria were extracted from literature review and a workshop with academics and practitioners. Hereinafter, let be $S = \{N$: not important, L : low importance, S : somewhat important, V : very important, E : extremely important$\}$. Figure 1 shows the linguistic expressions given by the 10 experts based on the linguistic term set of five elements S.



Sub-Criteria	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10
Home-Host Country Distance	V	N, L	V	V	N	L	V	L	N	S
Host country GDP per capita	L	L, S	L	S	L	L	S	L	N	S
Host country level of welfare state	L	S, V	V	L	L	S	S	L	N	S
Host country political stability perception	S	V	E	V	S	V	V	S	N	V
Host country's corruption perception	L	V	E	V	L	E	V	S	N	V
The city size	V	S	V	L	L	L	L	L	S	V
City's cultural and language distance perception	S	N	L, S	S	L	S	S	S	L	N
City's climate characteristics	N	V, E	N, L	E	S	L	V	S	S	L
City's connectivity—infrastructural features	V	L	S, V	V	L	V	V	S	N	S
City's reputation, image and prestige	S	L	S, V	S	S	S	S	L	N	S
City government degree of transparency	L	V	V, E	E	?	E	V	V	N	V
City government bureaucracy level	L	E	S, V	V	L	V	V	V	E	V
Access to financial support provided by city government	V	V	S, V, E	S	S	S	V	V	E	V
City government support to public-private partnerships (PPP)	V	E	V, E	S	?	S	V	V	V	V
City GDP per capita	S	S	S	L	?	L	S	L	N	V
Municipal economic budget	S	S, V	V, E	L	S	L	V	L	L, S	S
City R&D expenditure	S	S	S, V	L	L	L	N	L	L, S	V
The service economy of the city	S	L	L, S, V	V	S	V	V	L	L, S	S
Stakeholders' pressure in the city	S	S, V	V, E	V	L	S	V	S	L, S	V
Citizens' environmental awareness	L	V	V, E	E	E	S	L	L	L, S	V
City's air quality	L	V	S	S	V	S	N	L	L	V
Degree of city transition to renewables	L	V, E	S	E	N	L	V	V	E	V
Competition intensity in the city	V	S, V	V, E	L	L	S	S	L	N	S
Pool of skilled labor in the city	V	S, V	S, V, E	L	V	V	E	L	N, L	V
Access to needed suppliers	V	S	S, V, E	S	S	S	V	S	L, S	V
City's potential customers	V	V	V, E	E	V	V	S	S	S	E
City's degree of know-how, innovation and technological exchanges	S	L	S, V	S	E	S	N	L	S, V	V

Figure 1. Linguistic expressions given by the ten experts in relation to the importance of each sub-criteria [7].

Three different initial assumptions with respect to the type of perceptual-map owned by each of the ten experts involved in the group decision-aiding situation are considered. For each scenario, corresponding to each of these assumptions,

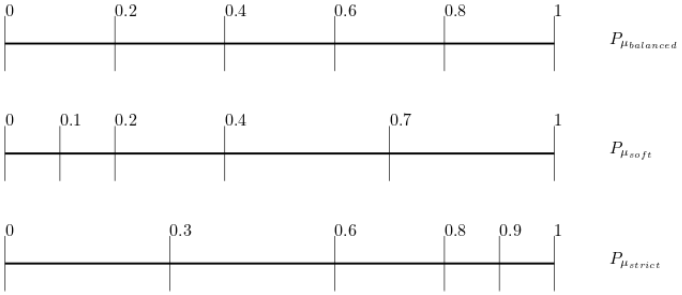


Figure 2. Partitions corresponding to the different perceptual maps assumed to model experts' opinions in situation A, situation B and situation C.

the evaluation results were computed. The three different assumptions (starting-points) based on the existence of different perceptual-maps within the group of experts are the following:

- **Situation A.** The setting with respect to the perceptual maps is assumed to be $\mu_l(s_i) = 0.2$, $\forall\ i \in \{1, 2, 3, 4, 5\}$ and $\forall\ l \in \{1, 2, \ldots, 10\}$. This means that the qualitative reasoning process of all experts can be modelled by means of an equally and symmetrically distributed LTS. The partition associated to this perceptual map, $\mu_{balanced}$, is shown in figure 2.
- **Situation B.** The setting with respect to the perceptual maps is assumed to be $\mu_l(s_1) = \mu_l(s_2) = 0.3$, $\mu_l(s_3) = 0.2$, $\mu_l(s_4) = \mu_l(s_5) = 0.1$, $\forall\ l \in \{1, 2, 3, 4, 5\}$ and $\mu_l(s_i) = 0.2\ \forall\ i \in \{1, 2, 3, 4, 5\}$, $\forall\ l \in \{6, 7, 8, 9, 10\}$. This represents a situation where the first five experts have a qualitative reasoning process that could be considered 'strict' or 'perfectionist', owing a perceptual map, μ_{strict}, illustrated in figure 2. The rest of the experts are assumed to elicit their opinions based on $\mu_{balanced}$.
- **Situation C.** The setting with respect to the perceptual maps is assumed to be $\mu_l(s_1) = \mu_l(s_2) = 0.1$, $\mu_l(s_3) = 0.2$, $\mu_l(s_4) = \mu_l(s_5) = 0.3$, $\forall\ l \in \{1, 2, 3, 4, 5\}$ and $\mu_l(s_i) = 0.2\ \forall\ i \in \{1, 2, 3, 4, 5\}$, $\forall\ l \in \{6, 7, 8, 9, 10\}$. In this situation, the first five experts are assumed to be 'generous' or 'soft' when eliciting their opinions. This is modeled with a perceptual map, μ_{soft}, whose associated partition is also illustrated in figure 2. The rest of the experts are assumed to elicit their opinions based on $\mu_{balanced}$.

Starting-points A, B and C are graphically illustrated in figure 3.

The projected LTS, S^* and the projected perceptual map, μ_* are computed for each situation:

- **Situation A.** The projected LTS is $S^*_A{=}\{s^*_1, s^*_2, s^*_3, s^*_4, s^*_5\}$ and $\mu^A_*{=}\mu^A_*(s^*_i) = 0.2\ \forall\ i \in \{1, 2, 3, 4, 5\}$.
- **Situation B.** The projected LTS is $S^*_B{=}\{s^*_1, s^*_2, s^*_3, s^*_4, s^*_5, s^*_6, s^*_7\}$ with $\mu^B_*{=}\mu^B_*(s^*_i) = 0.2\ \forall\ i \in \{1, 4, 5\}$ and $\mu^B_*{=}\mu^B_*(s^*_i) = 0.1\ \forall\ i \in \{2, 3, 6, 7\}$.
- **Situation C.** The projected LTS is $S^*_C{=}\{s^*_1, s^*_2, s^*_3, s^*_4, s^*_5, s^*_6, s^*_7\}$ with $\mu^C_*{=}\mu^C_*(s^*_i) = 0.2\ \forall\ i \in \{3, 4, 7\}$ and $\mu^C_*{=}\mu^C_*(s^*_i) = 0.1\ \forall\ i \in \{1, 2, 5, 6\}$.

The corresponding projected partitions are illustrated in figure 4.

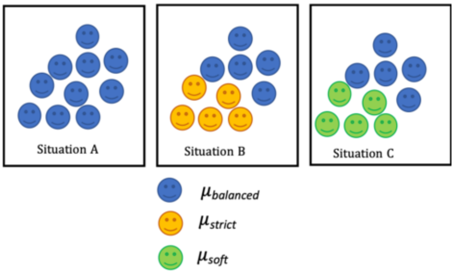


Figure 3. An illustration of the three different starting-points, based on the different perceptual-map hypothesis considered

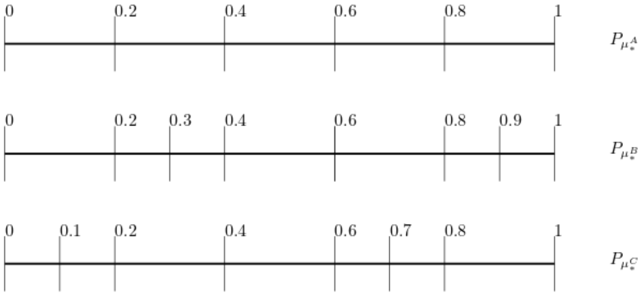


Figure 4. The projected partitions resulting of situation A, B and C

Then, for each situation A,B and C, using the projected perceptual map of Definition 5, each individual linguistic assessment is mapped onto the corresponding projected space.

The individual projected assessments are aggregated, for each sub-criterion, by considering a possibility function computed by normalizing the frequencies obtained from DMs' opinions.

Then, following an adapted version of the Fuzzy TOPSIS method [7,8], the positive and negative ideal solutions are identified, and the distances of each sub criteria to the positive and negative ideal solutions are calculated, respectively, by using the distance of Subsection 2.2. Based on these distances the closeness coefficient is obtained for each subcriteria. Based on the relative closeness coefficients' values, the partial weight of each sub-criterion within each criteria group is distributed, i.e., the weight percentages of each criteria group sum up to 100%. The subcriteria are then ranked within each criteria group. Combining the average weights of the main criteria with the relative importance of each sub-criteria within each group, a final ranking is obtained. The ranking is illustrated in Table 1.

According to the results of Table 1, the most relevant factor is 'City's potential customers', regardless of the perceptual-map hypothesis. Besides, the relevant

Sub-criteria	Sit. A	Sit. B	Sit.C
City?s potential customers	8.07%	8.59%	9.06%
Access to financial support provided by city government	7.96%	7.37%	7.76%
City government support to public-private partnerships	7.93%	7.20%	8.21%
City government degree of transparency	7.27%	8.26%	7.27%
City government bureaucracy level	6.84%	7.15%	6.75%
Stakeholders? pressure in the city	6.14%	4.03%	6.25%
Degree of city transition to renewables	6.09%	4.46%	5.24%
Access to needed suppliers	5.51%	4.22%	4.70%
Pool of skilled labor in the city	5.40%	5.41%	4.97%
Citizens? environmental awareness	4.10%	5.20%	5.60%
City?s degree of know-how, innovation and...	3.94%	3.54%	2.29%
Host country political stability perception	3.88%	2.87%	3.19%
The service economy of the city	3.86%	2.89%	2.78%
Host country?s corruption perception	3.44%	3.69%	3.18%
City?s air quality	2.81%	3.33%	2.15%
City?s connectivity?infrastructural features	2.75%	2.53%	3.04%
Home-Host Country Distance	2.29%	2.49%	2.30%
Municipal economic budget	2.09%	2.68%	2.39%
Competition intensity in the city	2.09%	3.22%	3.97%
City?s climate characteristics	1.69%	1.83%	2.24%
City?s reputation, image and prestige	1.52%	1.39%	1.07%
City GDP per capita	1.38%	2.33%	1.55%
City?s cultural and language distance perception	1.05%	0.63%	0.77%
The city size	1.00%	1.60%	0.86%
City R&D expenditure	0.52%	2.04%	1.00%
Host country level of welfare state	0.39%	0.93%	1.31%
Host country GDP per capita	0.00%	0.00%	0.00%

Table 1. Sub-criteria overall relative weight, for each situation considered

percentage weight of this sub-criterion is quite similar in the three situations: 8.0671 % , 8.590% and 9.06% for situations A, B and C, respectively. The rest of the sub-criteria related to market conditions for energy firms, which are access to needed suppliers, pool of skilled labor, city's degree of know-how and competition intensity in the city are placed in the 8th, 9th, 11th, 19th positions of the rank (situation A), respectively. In the case of situation B and C, these sub-criteria positions are: 9th, 6th, 12th, 14th and 10th, 9th, 18th and 11th, respectively. It is important to highlight that customers and suppliers' environments are more relevant than the factor of competition intensity in the city, regardless of the hypothesis.

As expected, the least valued sub-criterion is 'Host Country GDP per capita' in all situations.

It is also relevant to notice that the resulting TOP 5 sub-criterion are all the same, regardless of the hypothesis. Moreover, without considering the first ranked sub-criterion which belongs to market conditions, the following four sub-criteria all belong to the group of 'City's government and its policies'.

4. Conclusions

This paper proposes the concept of perceptual maps to model human opinions in a group decision-making context. The proposed approach considers a multi-granular projected structure using unbalanced hesitant linguistic term sets.This multi-perceptual framework allow us to deal with MCGDM problems where each decision maker has its own qualitative reasoning approach.

An illustrative case is presented in the location decisions made by multinationals enterprises of the energy sector within the European smart city context. The illustrative case is an extension of a previous study presented in [7] by considering different perceptual maps. In this multi-perceptual framework, criteria and sub-criteria governing the energy MNEs strategic location decisions are ranked.

Acknowledgments
This research was supported partly by the INVITE research project (TIN2016-80049-C2-1-R and TIN2016-80049-C2-2-R), funded by the Spanish Ministry of Science and Information Technology and the European Union Horizon 2020 Research and Innovation Programme, under the grant agreement No 731297.

References

[1] Agell, N. et al., 2012. Ranking multi-attribute alternatives on the basis of linguistic labels in group decisions. *Information Sciences* 209, 49-60.

[2] Chiclana, F. et al. Type-1 OWA Unbalanced Fuzzy Linguistic Aggregation Methodology: Application to Eurobonds Credit Risk Evaluation. *International Journal of Intelligent Systems* 33.5 (2018), 1071-1088.

[3] Fu, C. and Yang, S-L. The group consensus based evidential reasoning approach for multiple attributive group decision analysis. textitEuropean Journal of Operational Research 206.3 (2010), pp. 601-608.

[4] Chen, Z-S. et al. Identifying and prioritizing factors affecting in-cabin passenger comfort on high-speed rail in China: A fuzzy-based linguistic approach. textitApplied Soft Computing 95 (2020), 106558.

[5] Liao, H., Xu, Z. and Zeng, X.J. Distance and similarity measures for hesitant fuzzy linguistic term sets and their application in multi-criteria decision making, *Information Sciences* 271 (2014) 125–142.

[6] Montserrat-Adell, J., Agell, N., Sánchez, M., Prats, F. and Ruiz, F.J. Modeling groups assessments by means of hesitant fuzzy linguistic term sets. *Journal of Applied Logic.* Accepted to be published.

[7] Porro, O., Pardo-Bosch, F., Agell, N., Sánchez, M. Understanding Location Decisions of Energy Multinational Enterprises within the European Smart Cities? Context: An Integrated AHP and Extended Fuzzy Linguistic TOPSIS Method. *Energies*, 13(10) (2020), 2415.

[8] Porro, O. *A hesitant fuzzy perceptual-based approach to model linguistic assessments*, PhD Thesis. UPC BarcelonaTech (2021).

[9] Porro, O., Agell, N., Sánchez, M. Ruiz, F.J. A multi-attribute group decision model based on unbalanced and multi-granular linguistic information: An application to assess entrepreneurial competencies in secondary schools, *Applied Soft Computing*, 111 (2021) 107662.

[10] Rodríguez, R.M., Martínez, L., Herrera, F. 2012. Hesitant fuzzy linguistic terms sets for decision making, *IEEE Transactions on Fuzzy Systems* 20 (1), 109-119.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210115

On the Temperature of SAT Formulas

Jesús GIRÁLDEZ-CRU [a,1], Pedro ALMAGRO-BLANCO [b]
[a] *DaSCI, DECSAI, Universidad de Granada (jgiraldez@ugr.es)*
[b] *Dpto. Ciencias de la Computación e Inteligencia Artificial, Universidad de Sevilla (palmagro@us.es)*

Abstract.

The remarkable advances in SAT solving achieved in the last years have allowed to use this technology in many real-world applications of Artificial Intelligence, such as planning, formal verification, and scheduling, among others. Interestingly, these industrial SAT problems are commonly believed to be easier than classical random SAT formulas, but estimating their actual hardness is still a very challenging question, which in some cases even requires to solve them. In this context, realistic pseudo-industrial random SAT generators have emerged with the aim of reproducing the main features shared by the majority of these application problems. The study of these models may help to better understand the success of those SAT solving techniques and possibly improve them.

In this work, we present a model to estimate the *temperature* of real-world SAT instances. This temperature represents the degree of distortion into the expected structure of the formula, from highly structured benchmarks (more similar to real-world SAT instances) to the complete absence of structure (observed in the classical random SAT model). Our solution is based on the Popularity-Similarity (PS) random model for SAT, which has been recently presented to reproduce two crucial features of application SAT benchmarks: scale-free and community structures. The PS model is able to control the hardness of the generated formula by introducing some randomizations in the expected structure. Our solution is a first step towards a hardness oracle based on the temperature of SAT formulas, which may be able to estimate the cost of solving real-world SAT instances without solving them.

Keywords. SAT, hardness, temperature, Popularity-Similarity, entropy

1. Introduction

Despite the remarkable progress in SAT solving techniques in the last years, determining the time required to solve a given SAT instance by a certain algorithm is still today one of the most interesting and challenging questions in the SAT community. The simplest solution is to run that algorithm until termination, but unfortunately this task may be extremely costly, and hence infeasible. An alternative solution would be to *accurately estimate* its solving time.

Most of the traditional approaches on the study of the hardness of SAT instances have focused on the so-known classical random model of SAT formulas [28], where a

[1] Corresponding author. Supported through the Juan de la Cierva program (FJCI-2017-32420, IJC2019-040489-I).

random formula $F_k(n,m)$ is a set of m clauses over n variables, and clauses are chosen uniformly and independently among all the $2^k\binom{n}{k}$ non-trivial clauses of length k.[2] The empirical hardness of this model has been extensively studied [28,35]. In particular, for any fixed n and $k > 2$, there exists an easy-hard-easy pattern depending on the clause/variable ratio m/n, which is also related to the satisfiability of the formula. Therefore, the hardness of random SAT formulas simply depends on k, n and m. The natural question is whether a *simple* hardness characterization also exists for real-world SAT instances, which is the question that motivates our work. Far from providing such a characterization, in this work we analyze the relation between the hardness of real-world SAT instances and a simple parameter of them, as a first step toward facing this challenge.

Although the reasons of the success of CDCL SAT solvers on the heterogeneous set of application SAT instances are not completely understood yet [34,6], there have been some recent attempts to study common features on these industrial problems [1] with the aim of explaining the good performance of these solvers on this benchmark. Because of this heterogeneity, realistic pseudo-industrial random SAT instances generators have emerged, stated as one of the most important challenges in propositional search [32]. The cornerstone of these models is to produce random formulas with computational properties similar to real-world instances. The Popularity-Similarity (PS) random model [20] has been proposed as one of these realistic random SAT generators.

The PS model defines an expected structure composed of scale-free structure [3] (the number of variables occurrences follows a power-law distribution, i.e., a few variables occurs a lot whilst most of them occur very little) and community structure [5] as a result of high clustering (the set of variables can be split into disjoint communities such that variables mostly occur in clauses with other variables of the same community). They are two common features in most real-world SAT benchmarks. Inspired by the *entropy* of physical systems, the PS model uses the *temperature* T as a parameter to control the degree of distortion into this structure. This is, at $T = 0$ the model produces that structure with high probability (hence the generated formula is more similar to real-world SAT instances), whereas at high temperature the model behaves like the classical random SAT model (hence the generated formula does not exhibit any structure at all).

In practice, it has been observed that CDCL solvers focus on frequent variables and on variables of the same community [6,4,19,7,2]. Using the synthetic PS model, it has been also observed that CDCL solvers perform better on PS formulas with low temperature. On the contrary, SAT solvers specialized in classical random SAT formulas perform better on PS formulas with high temperature [20]. We conjecture that the hardness of real-world SAT formulas depends on a notion of temperature, which characterizes the distortion into the structure of a particular formula from the structure exhibited in most real-world SAT benchmarks. To this end, we consider a real-world SAT problem as an instantiation of the PS model [20], and its temperature corresponds to the value of T in this instantiation. We emphasize that this does not require the real-world instance to have any structure (e.g., high T).

In order to test our hypothesis, we need first to compute the temperature of a given SAT formula. Unfortunately, there is no known analytical method to this purpose [31]. In our work we present an extensive study of Machine Learning (ML) regression methods to estimate it. In particular, we analyze the performance of different ML techniques trained

[2] A non-trivial clause of length k contains k distinct, non-complementary literals.

with PS formulas generated at distinct temperatures, and measure their accuracy in the estimation. We also evaluate the robustness of each ML technique when the training set is altered with perturbations in the generation step. Empirically, we show that ML techniques based on ensembles (e.g., random forest) are the most accurate approaches, and they remain robust to perturbations.

2. Related Work

There are in the literature other realistic SAT generators, such as the scale-free SAT model [4], which generates purely scale-free SAT instances, and the community attachment model [18], which is able to produce formulas with clear community structure. We recall that both features can be observed in PS formulas. The hardness of scale-free SAT formulas [4] also depends on the exponent β of the power-law distribution that characterizes them [15,16,30].

A seminal contribution on ML applied to SAT solving is SATzilla [36,23]. A SAT solver unlikely dominates all others on unrestricted SAT instances, but it may show a particularly good performance on a certain class of problems [26]. On this idea, SATzilla proposes a per-instance algorithm portfolio that estimates the best solver to solve a given formula from a predefined set. This portfolio approach has also been successfully applied in other works [24,27].

3. The PS Random SAT Model

In this section we provide a brief description of the Popularity-Similarity (PS) random SAT model [20]. For further details, we address the reader to the original reference.

The PS model is able to generate random SAT formulas with both scale-free and community structure as the result of two orthogonal forces: popularity and similarity. To model them, every variable i is assigned radial and angular coordinates r_i and θ_i, representing respectively its popularity and its similarity to other variables. Popular variables have a small radius and similar variables have close angles. These two coordinates are also assigned to every clause j. In this model, the probability $P(i \leftrightarrow j)$ of a variable i occurring in a clause j (with any sign) is:

$$P(i \leftrightarrow j) = \left(1 + \left(\frac{r_i^{\beta} \cdot r'^{\beta'}_j \cdot \theta_{ij}}{R}\right)^{1/T}\right)^{-1}$$

where r_i and r_j represent the radiuses of i and j respectively, θ_{ij} is the angular distance between them, β and β' are respectively the exponents of the power-law distributions for variables occurrences and clauses length, R is a normalization constant ensuring the expected formula size, and T is the temperature of the model. Therefore, the temperature T precisely controls the entropy of the system, i.e., the degree of distortion into the expected probabilities. The aforementioned structures are the result of this probability distribution, which is clearly non-uniform at low T: it is more likely that a clause j contains a popular variable (low r_i) or a variable similar to it (low θ_{ij}). In contrast, the probability distribution becomes (close to) uniform for high values of T, as in the classical random SAT model.

4. ML-based Regression Techniques

In this section, we provide a general overview of the regression problem to solve, and the techniques we use for that task.

Let us consider an instance ϕ, which is characterized by a vector $\mathbf{x}_\phi = [x_\phi^1, \dots, x_\phi^n]$ of n features. Being $x^* \notin \mathbf{x}$ the target feature to estimate, the problem consists of finding the function f s.t. $f(\mathbf{x}) = x^* \pm \varepsilon$ that minimizes ε. In our case, ϕ represents a SAT instance, $\mathbf{x}$ its features, and x^* its temperature T. Since the temperatures of the PS instances used in the training step are known *a priori*, we use supervised ML techniques to estimate f. In the following section, it is discussed the set of features $\mathbf{x}$ used in our experiments.

In our problem, the target estimation x^* is a continuous value. Although there exist many different ML algorithms to predict these values, the *no-free-lunch theorem* [21] states that it cannot be known *a priori* which techniques show a good performance in a particular problem, and finding them usually requires a trial and error process.

In our experimental analysis, we evaluate a total of 13 distinct regression methods. They can be grouped into the following categories. (1) *Linear regression*: LR (ordinary least squares linear regression), SGD (Stochastic Gradient Descent), PassiveAgressive [10], RANSAC (Random Sample Consensus) [11], Theil-Sen (TS) [11], Huber [22], and Bayesian Ridge [33]. (2) *Ensemble methods*: RandomForest (RF) [8], ExtraTrees (ET) [17], AdaBoost (AB) [14], and XGBoost (XGB) [9]. (3) *Neural networks*: FNN (Feed-forward Neural Network) with one hidden layer, 64 hidden neurons, and *Adam* optimizer [25]. (4) *Other methods*: K-neighbors. For all these methods, we use the implementation in sklearn [12].

For space constraints, we only give a general overview of these techniques. Linear regression methods will help us to evaluate possible linear relations between the set of features used to represent a formula and its temperature. Ensemble methods construct a set of *weak regressors* and aggregate their predictions reducing the overfitting that would be obtained when applying the regressors individually. It is well known that a feed-forward network with a single hidden layer containing a finite number of neurons can approximate continuous functions. Due to the already good performance achieved by the FNN used in our experiments, we have not considered more complex FNN, or even other graph neural network architectures. However, it would be interesting to study them and analyze their performance in future works.

In our experiments, we also use Bayesian optimization [29] to tune the hyperparameters of the ML techniques in order to increase their accuracy. In order to evaluate each regression model, we perform a 10-folds cross-validation.

5. Analysis of the Temperature Estimation

In this section, we present an exhaustive experimental analysis of regression techniques to estimate the temperature of SAT instances.

5.1. Experimental Set-Up

Generation of SAT formulas. The training set is composed of an heterogeneous set of PS random SAT formulas, differing in their number of variables $n \in [100 \dots 5000]$, and their clause/variables densities $m/n \in [2 \dots 8]$. For each value of n and m/n, we generate 100 random PS formulas with distinct temperatures. Our main benchmark results into a total of 7200 SAT instances, containing both satisfiable and unsatisfiable formulas. All formulas are 3-CNF and much smaller than real-world SAT instances. However, we found experimentally that the formula size has no impact on the performance of ML methods.

The popularity and similarity of the generated formulas is controlled by the parameter β. In the main batch of experiments, we use $\beta = 1$, i.e., a very clear scale-free structure. We also evaluate the robustness of the regression techniques exposing the training set to some perturbations on β (see Section 5.3.1)

Values of Temperature. In the PS model, a small difference of T at low temperatures may result in major differences in the resulting structure of the generated formula, whereas at high temperatures this structure is almost unaltered. For this reason, instead of working directly with the temperature, we sample and estimate its logarithm. In particular, for each formula size we sample 100 uniformly distributed random values in the interval $[-1, 1]$. These values correspond to the logarithm of the temperature of the generated formulas, and they are the values used to train the regression models and to be estimated. Therefore, the actual temperature ranges in the interval $[1/e, e]$. Although this generates a reasonable range of temperatures, we also evaluate some perturbations on this interval in Section 5.3.2.

Set of features. Every SAT instance is characterized by a vector of features. Ideally, this vector contains a set of uncorrelated, fast-to-compute features of the formula. In our experiments, we use the extended and well-known set of features used in the SATzilla toolkit [36]. In particular, we use a total of 101 features,[3] including formula size, graph characteristics, and solver statistics. In Section 5.4 we evaluate the impact of reducing the number of features used to train the regression models, in order to analyze their feature importance.

Filtering out trivial instances. Random PS SAT instances at low temperatures may be very easy [20]. This is especially relevant in small formulas (e.g., $n = 100$), which might be even solved by simple preprocessing techniques. For this reason, we filter out those trivial instances from the benchmark because some SATzilla features include solver statistics. We observed that the resulting unbalance does not affect the performance of the regression models with the best accuracy, due to the already large number of formulas in the benchmark.

Accuracy of the model. In order to evaluate each regression technique, we use the well-known coefficient of determination R^2 between the actual temperature and its prediction. Recall that $R^2 \in [-\infty, 1]$ with positive values indicating the existence of a certain correlation between the observed data and their predictions (the higher the value of R^2, the better the prediction).

In our experiments we use the value $R^2 \geq 0.8$ to distinguish those regression methods achieving a strong correlation (i.e., a good accuracy in the prediction), although any other

[3] We skip the computation of LP-based and SLS-based features due to their long execution time for some formulas. We also skip diameter features, as done in the last SATzilla version.

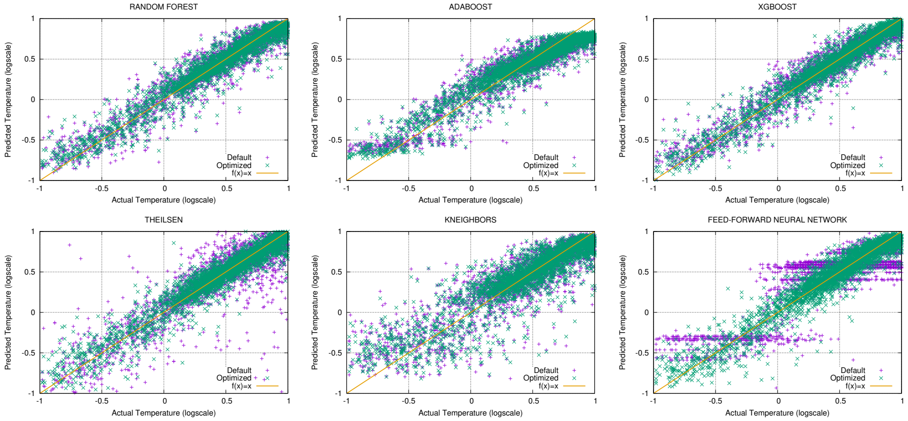


Figure 1. Predicted temperature versus actual temperature, for some regression techniques with a small standard deviation.

reasonable high value of R^2 could have been used instead. It is worth noticing that all methods with such a strong correlation also show a very small standard deviation of R^2, always lower than 0.15 (and usually much lower).

5.2. Performance of Regression Methods

The first natural question in our analysis is whether a regression method is able to estimate the temperature of a given PS SAT formula given the vector of SATzilla features for this formula.

In Table 1, we summarize the performance of each regression method on this problem, with both default and optimized hyperparameter settings, measuring the average and standard deviation of the coefficient of determination R^2.

Regression Method	Default	Optimized
LR	0.789 ± 0.35	0.789 ± 0.35
SGD	-1.230 ± 2.24	-2.5e8 ± 3e8
PassiveAggressive	-181.7 ± 271	-1.072 ± 0.13
RANSAC	-0.982 ± 1.53	0.511 ± 1.20
TS	-8.272 ± 25.3	**0.898 ± 0.03**
Huber	-0.085 ± 0.14	-0.076 ± 0.15
BayesianRidge	0.760 ± 0.46	0.694 ± 0.66
RF	**0.921 ± 0.01**	**0.931 ± 0.01**
ET	**0.922 ± 0.01**	**0.935 ± 0.01**
AB	**0.878 ± 0.02**	**0.896 ± 0.01**
XGB	**0.924 ± 0.01**	**0.935 ± 0.00**
FNN	0.687 ± 0.07	**0.912 ± 0.01**
Kneighbors	0.795 ± 0.03	0.796 ± 0.02

Table 1. Avg. coefficient of determination R^2 for different regression methods, with default and optimized hyperparameters.

We can observe that many of the methods with default settings are able of predicting the temperature of the formulas with a high accuracy, hence showing the robustness of

	Baseline	Perturbations on β				Perturbations on T interval	
β	$\beta = 1$	$\beta = 5/6$	$\beta = 4/6$	$\beta = 3/6$	Union(β)		
$log T$	$[-1,1]$					$[-2,2]$	$[-3,3]$
LR	0.789 ± 0.35	-3.199 ± 12.4	-0.568 ± 4.45	**0.856 ± 0.13**	0.785 ± 0.02	**0.936 ± 0.02**	**0.907 ± 0.01**
TS	**0.898 ± 0.03**	-2.223 ± 9.40	-6.625 ± 22.3	-197.1 ± 241	-213.6 ± 270	**0.939 ± 0.01**	**0.904 ± 0.02**
RF	**0.931 ± 0.01**	**0.932 ± 0.01**	**0.942 ± 0.01**	**0.958 ± 0.00**	**0.858 ± 0.01**	**0.955 ± 0.01**	**0.945 ± 0.01**
ET	**0.935 ± 0.01**	**0.935 ± 0.01**	**0.946 ± 0.01**	**0.960 ± 0.01**	**0.861 ± 0.01**	**0.956 ± 0.01**	**0.947 ± 0.01**
AB	**0.896 ± 0.01**	**0.882 ± 0.01**	**0.883 ± 0.01**	**0.904 ± 0.01**	0.678 ± 0.01	**0.928 ± 0.00**	**0.904 ± 0.01**
XGB	**0.935 ± 0.00**	**0.937 ± 0.01**	**0.948 ± 0.01**	**0.963 ± 0.01**	**0.872 ± 0.01**	**0.956 ± 0.01**	**0.943 ± 0.01**
FNN	**0.912 ± 0.01**	**0.904 ± 0.02**	**0.915 ± 0.01**	**0.927 ± 0.01**	0.780 ± 0.02	**0.917 ± 0.01**	**0.874 ± 0.02**

Table 2. Avg. coefficient of determination R^2 for different regression methods (with optimized hyperparameter), varying the training data in: (i) the value of β, and (ii) the interval of the temperatures.

our approach. Those accurate methods are based on ensembles, which are commonly more robust to hyperparameter tuning. FNN and Kneighbors present an acceptable accuracy with a low standard deviation. Although models LinReg and BayesianRidge also obtain an acceptable R^2 average, we cannot draw definitive conclusions due to their high deviation.

After optimizing hyperparemeters, we observe noticeable improvements in most of the methods, especially in FNN and TheilSen. In the case of FNN, the method shows a considerable improvement after adjusting the number of neurons in the hidden layer, the optimizer and the batch size. In the case of TheilSen, this linear method is able to learn from different subsets of the training data, and with an optimal configuration acquires resistance against outliers. This fact suggests a certain linear relation between SAT features and the temperature T. It is worth noticing that, in general, linear methods do not outperform (non-linear) methods based on neural networks and ensembles.

In Figure 1, we depict the predicted temperature versus the actual temperature of random PS formulas for six regression techniques. Notice that when R^2 is close to 1, the points on the figure must be close to the diagonal.

5.3. Robustness to Benchmark Perturbations

Regression Met.	All (baseline)	Basic (A)	Graph (B)	CDCL (C)	Union $A \cup B \cup C$
LinReg	0.789 ± 0.35	0.118 ± 2.14	**0.808 ± 0.08**	0.330± 0.07	0.749± 0.41
TheilSen	**0.898 ± 0.03**	-0.058 ± 2.62	**0.832 ± 0.03**	0.336± 0.06	-0.143± 3.11
RandomForest	**0.931 ± 0.01**	**0.911 ± 0.01**	**0.917 ± 0.01**	0.579± 0.04	**0.927± 0.01**
ExtraTrees	**0.935 ± 0.01**	**0.918 ± 0.01**	**0.924 ± 0.01**	0.565± 0.05	**0.933± 0.01**
AdaBoost	**0.896 ± 0.01**	**0.863 ± 0.02**	**0.864 ± 0.01**	0.331± 0.06	**0.876± 0.02**
XGBoost	**0.935 ± 0.00**	**0.916 ± 0.01**	**0.923 ± 0.02**	0.554± 0.02	**0.932± 0.01**
FNN	**0.912 ± 0.01**	**0.889 ± 0.01**	**0.932 ± 0.02**	0.535± 0.06	**0.911± 0.01**

Table 3. Avg. coefficient of determination R^2 for different regression methods (with optimized hyperparameter), for different features sets.

The next natural question is whether the accuracy of our approach is robust to perturbations. In particular, we consider perturbations in the training step modifying the parameters values of the generated PS random SAT formulas varying: (i) the scale-free structure of the benchmark, and (ii) the interval used to sample the values of the temperature.

5.3.1. Varying the Scale-Free Structure

In our main benchmark, all PS random formulas are generated with $\beta = 1$, i.e., with a clear scale-free structure. Now we analyze the performance robustness of the regression models in benchmarks just differing in the value of β used in the generation of the SAT formulas. In particular, we evaluate the cases with $\beta = \{5/6, 2/3, 1/2\}$, and the union of these four. In this experiment, we use the regression models with optimized parameters computed for $\beta = 1$. We restrict our analysis to the regression methods that already showed a good performance in the previous experiment, and adding LinReg as baseline. In Table 2, we summarize the results of this experiment.

We observe that, in all cases, the regression methods showing the best performance in all benchmarks (with any value of β) are techniques that already showed a very good performance with default parameters in the benchmark with $\beta = 1$, i.e., (non-linear) methods based on ensembles of decision trees: RandomForest, ExtraTrees and XGBoost. Therefore, these techniques seem to be robust to this perturbation, and hence they are good candidates to build a promising temperature estimator. Surprisingly, the LinReg method is able to obtain a good result in the case of the union of the different sets. This can be due to the fact that this set of instances is larger than the others, allowing the linear method to learn more effectively. In the case of TheilSen, the optimization made for $\beta = 1$ does not generalize correctly, thus this model is not robust.

5.3.2. Varying the Interval of Temperatures

Another perturbation to study the robustness of the regression methods is the interval used to sample the values of the temperature of the random PS formulas in the benchmark. We recall these values represent the logarithm of the temperature of the generated formulas. In the main benchmark, we use the interval $[-1, 1]$ (i.e., the temperature ranges in $[1/e, e]$). In this experiment, we generate two similar benchmarks only differing in this interval: $[-2, 2]$ and $[-3, 3]$. In Table 2 we also summarize the results of this perturbation.

We observe that all regression methods show a very good accuracy, suggesting that they all are robust to this kind of perturbation in the temperature. Interestingly, there seems to be a peak of performance in the intermediate interval $[-2, 2]$, i.e., using a relatively ample interval is beneficial, but using a too ample one is not.

5.4. Features Set and Feature Importance

In this section we analyze the relation between the set of features used by the regression techniques and their predictive capacity.

The set of features provided by the SATzilla toolkit can be divided into the following categories: (a) Basic features, (b) Graph features, (c) CDCL (and DPLL) features, and (d) other solving and timing statistics. In our analysis, we focus on the first three subsets, which respectively have 26, 25, and 24 features. We use the original SATzilla tool to produce all these features [36].

In Table 3, we summarize the coefficient of determination R^2 of different regression methods using a different set of features to train the models and compute the regression. Again, we use the models with optimized hyperparameter settings from the main experiment (see Table 1).

We observe that the set of features used to compute the regression may have dramatic consequences in its performance. In particular, we observe a very poor performance when only CDCL features are used. On the contrary, the performance is generally good using the set of Graph features. Surprisingly, the linear methods (LinReg and TheilSen) have very good performance when they are trained using Graphs features only, whilst their performance gets worse with the rest of the feature sets. This shows that these graph characteristics are able to *linearize* the relation between the SAT formula and its temperature. Linear estimators are less sensitive to overfitting than others. Therefore, the combination between this small set of features and these linear methods must be emphasized.

In the case of CDCL features, the determination coefficient R^2 of linear methods is similar to the one of the remaining non-linear regression techniques. This suggests a second linear relation between the set of features and the temperature. Nevertheless, it is much weaker. Basic features only produces good results with non-linear regression methods based on neural networks and ensembles, and this may explain the worse performance of linear methods when using them.

6. Conclusions

In this work, we have presented an extensive analysis of ML regression techniques in order to estimate the temperature of real-world SAT instances. Our experimental results show that ML methods based on ensembles (e.g., random forest) show the best performance, remaining robust to perturbations in the training step. Additionally, we have showed that a successful application like SATzilla is able to indirectly infer the temperature of the formulas using only a subset of (graph) features. As future work, we plan to extend this analysis with more sophisticated neural networks, including graph neural networks, and other automated ML techniques [13], and use these regression methods to estimate the temperature of real-world SAT instances.

References

[1] C. Ansótegui, M.L. Bonet, J. Giráldez-Cru, and J. Levy. Structure features for SAT instances classification. *Journal of Applied Logic*, 23:27–39, 2017.

[2] C. Ansótegui, M.L. Bonet, J. Giráldez-Cru, J. Levy, and L. Simon. Community structure in industrial SAT instances. *Journal of Artificial Intelligence Research*, 66:443–472, 2019.

[3] C. Ansótegui, M.L. Bonet, and J. Levy. On the structure of industrial SAT instances. In *Proc. of CP 2009*, pages 127–141, 2009.

[4] C. Ansótegui, M.L. Bonet, and J. Levy. Towards industrial-like random SAT instances. In *Proc. of IJCAI 2009*, pages 387–392, 2009.

[5] C. Ansótegui, J. Giráldez-Cru, and J. Levy. The community structure of SAT formulas. In *Proc. of SAT 2012*, pages 410–423, 2012.

[6] G. Audemard and L. Simon. Predicting learnt clauses quality in modern SAT solvers. In *Proc. of IJCAI 2009*, pages 399–404, 2009.

[7] G. Baud-Berthier, J. Giráldez-Cru, and L. Simon. On the community structure of bounded model checking SAT problems. In *Proc. of SAT 2017*, pages 65–82, 2017.

[8] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[9] T. Chen and C. Guestrin. XGBoost: A scalable tree boosting system. In *Proc. of KDD 2016*, pages 785–794, 2016.

[10] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer. Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 7(Mar):551–585, 2006.

[11] X. Dang, H. Peng, X. Wang, and H. Zhang. Theil-sen estimators in a multiple linear regression model. *Olemiss Edu*, 2008.

[12] L. Buitinck et al. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

[13] M. Feurer, A. Klein, K. Eggensperger, J.T. Springenberg, M. Blum, and F. Hutter. Auto-sklearn: Efficient and robust automated machine learning. In *Automated Machine Learning - Methods, Systems, Challenges*, pages 113–134. Springer, 2019.

[14] Y. Freund and R.E. Schapire. A decision-theoretic generalization of online learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[15] T. Friedrich, A. Krohmer, R. Rothenberger, and A.M. Sutton. Phase transition for sclae-free SAT formulas. In *Proc. of AAAI 2017*, pages 3893–3899, 2017.

[16] T. Friedrich and R. Rothenberger. Sharpness of the satisfiability threshold for non-uniform random k-SAT. In *Proc. of SAT 2018*, pages 273–291, 2018.

[17] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine Learning*, 63(1):3–42, 2006.

[18] J. Giráldez-Cru and J. Levy. Generating SAT instances with community structure. *Artificial Intelligence*, 238:119–134, 2016.

[19] J. Giráldez-Cru and J. Levy. Locality in random SAT instances. In *Proc. of IJCAI 2017*, pages 638–644, 2017.

[20] J. Giráldez-Cru and J. Levy. Popularity-similarity random SAT formulas. *Artificial Intelligence*, 299:103537, 2021.

[21] Y. Ho and D.L. Pepyne. Simple explanation of the no-free-lunch theorem and its implications. *Journal of Optimization Theory and Applications*, 115(3):549–570, 2002.

[22] P.J. Huber. *Robust statistics*. Springer, 2011.

[23] F. Hutter, L. Xu, H.H. Hoos, and K. Leyton-Brown. Algorithm runtime prediction: Methods & evaluation. *Artificial Intelligence*, 206:79–111, 2014.

[24] S. Kadioglu, Y. Malitsky, M. Sellmann, and K. Tierney. ISAC - instance-specific algorithm configuration. In *Proc. of ECAI 2010*, pages 751–756, 2010.

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

[26] K. Leyton-Brown, H.H. Hoos, F. Hutter, and L. Xu. Understanding the empirical hardness of *NP*-complete problems. *Communications of the ACM*, 57(5):98–107, 2014.

[27] Y. Malitsky, A. Sabharwal, H. Samulowitz, and M. Sellmann. Non-model-based algorithm portfolios for SAT. In *Proc. of SAT 2011*, pages 369–370, 2011.

[28] D.G. Mitchell, B. Selman, and H.J. Levesque. Hard and easy distributions of SAT problems. In *Proc. of AAAI 1992*, pages 459–465, 1992.

[29] J. Mockus. *Bayesian approach to global optimization: theory and applications*, volume 37. Springer Science & Business Media, 2012.

[30] O. Omelchenko and A. Bulatov. Satisfiability threshold for power law random 2-SAT in configuration model. *CoRR*, abs/1905.04827, 2019.

[31] F. Papadopoulos, M. Kitsak, M.A. Serrano, M. Boguñá, and D. Krioukov. Popularity versus similarity in growing networks. *Nature*, 489:537–540, 2012.

[32] B. Selman, H.A. Kautz, and D.A. McAllester. Ten challenges in propositional reasoning and search. In *Proc. of IJCAI 1997*, pages 50–54, 1997.

[33] M.E. Tipping. Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1(Jun):211–244, 2001.

[34] R. Williams, C.P. Gomes, and B. Selman. Backdoors to typical case complexity. In *Proc. of IJCAI 2003*, pages 1173–1178, 2003.

[35] L. Xu, H.H. Hoos, and K. Leyton-Brown. Predicting satisfiability at the phase transition. In *Proc. of AAAI 2012*, 2012.

[36] L. Xu, F. Hutter, H.H. Hoos, and K. Leyton-Brown. Satzilla: Portfolio-based algorithm selection for SAT. *Journal of Artificial Intelligence Research*, 32:565–606, 2008.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210116

Choosing the Root of the Tree Decomposition When Solving WCSPs: Preliminary Results

Aleksandra PETROVA [a,1], Javier LARROSA [a] and Emma ROLLON [a]
[a] *Dept. of Computer Science, UPC, Barcelona, Spain*
(petrova, larrosa, erollon)@cs.upc.edu

Abstract. In this paper we analyze the effect of selecting the root in a tree decomposition when using decomposition-based backtracking algorithms. We focus on optimization tasks for Graphical Models using the BTD algorithm. We show that the choice of the root typically has a dramatic effect in the solving performance. Then we investigate different simple measures to predict near optimal roots. Our study shows that correlations are often low, so the automatic selection of a near optimal root will require more sophisticated techniques.

Keywords. Weighted CSPs, Tree Decomposition, Discrete Optimization

1. Introduction

Many combinatorial optimization tasks on graphs can be computed in linear time when the graph is acyclic. It is well-known that in many cases we can benefit from this observation even when the graph of interest is cyclic by using a so-called *tree decomposition* which maps the graph into a tree [1]. Roughly speaking, the nodes in the decomposition contain the cyclic parts of the graph and the edges capture the acyclic interaction between the cyclic parts. Then, the tree decomposition can be used to guide a dynamic programming algorithm that proceeds bottom-up in the decomposition solving the respective subproblems. Typically such algorithms have time and space complexity exponential in the size of the largest cyclic part, which is called the (tree decomposition) *width* [2].

This approach has been widely studied in the context of *Graphical Models* [3], which is an umbrella term that covers a broad number of problems such as *Bayesian Networks*, *Markov Networks* [4], *Weighted CSPs* [5,3], etc. If we can find a small tree width for the instance of interest, then we can apply directly a dynamic programming algorithm (e.g. Bucket Elimination [6,7]) and solve it very efficiency. However, if we can only find decompositions with large width, then dynamic programming is useless (typically for its space complexity) and we have to rely on heuristic search methods such as Depth-first or Best-first. Heuristic search algorithms are unfeasible in terms of worst-case complexity because they are exponential in the problems size. However, the Graphical Models com-

[1] Projects RTI2018-094403-B-C33, funded by: FEDER/Ministerio de Ciencia e Innovacin Agencia Estatal de Investigacin,Spain

munity has enhanced them with intelligent pruning techniques and they can solve many large instances in pretty reasonable time [8].

A very fruitful line of research is the exploitation of decomposition trees beyond small widths by adapting heuristic search to the problem decomposition. The basic idea is simple and has been known for decades as AND/OR search [9]. In the context of Graphical Models corresponds to moving from a tabular to a memoization implementation of DP. However, the real challenge is to make it work in practice, that is, to add all the advantageous pruning techniques developed for standard heuristic search [10,11,12]. Such algorithms, that we call decomposition-based backtracking algorithms, have a proven time and space worst-case complexity exponential in the tree width, but their actual performance is much better than that due to the pruning techniques.

The definition of decomposition tree is not rooted, but decomposition-based algorithms must pick one. For standard implementations of dynamic programming algorithms such choice is not very important, since worst-case complexity is a tight bound of average-case complexity. However, some authors have reported that in decomposition-based backtracking algorithms choosing the root has an impact in performance [13].

In this paper we want to dig into this issue. We focus on the *backtracking tree-decomposition* algorithm BTD [10] inside toulbar2 [2] which is arguably the best example of this line of work.

We report an experiment in which we solve a carefully selected set of instances with all clusters as root. The experiment confirms that the impact of choosing a good root is high. We also observe that such impact changes very much from one problem to another. In some instances the best root hardly halves the time of the worst root. However, in other instances the best root makes the algorithm thousands of times faster than the expected time of making a random choice. The results from the experiment raise the natural question of how to identify a good root from the structure of the decomposition. To answer that question we have measured the correlation between performance and some simple yet reasonable cluster parameters. We observed that such simple parameters are often uncorrelated to performance, so they cannot be used as a heuristic way to identify a near optimal root.

To the best of our knowledge, this is the first systematic study on the impact of root selection in decomposition-based backtracking algorithms for Graphical Model optimization tasks, and the first attempt to identify a predictor of near optimal roots. Since we have not found simple single measurements correlated to algorithmic performance, we hypothesize that more sophisticated techniques such as machine learning are needed.

2. Preliminaries

2.1. Graphical Models

A *Graphical Model* is a tuple $P = \{X, D, F\}$ where $X = \{x_1, \ldots, x_n\}$ is the set of *variables*, $D = \{d_1, \ldots, d_n\}$ is the set of *domains* (d_i is the domain of x_i), F is the set of *cost functions*. Each cost function $f_S \in F$ has associated a subset of variables $S \subseteq X$, called *scope*, and the function assigns a cost to each possible assignment of these variables. A Graphical Model implicitly specifies an objective function

[2] https://miat.inrae.fr/toulbar2/

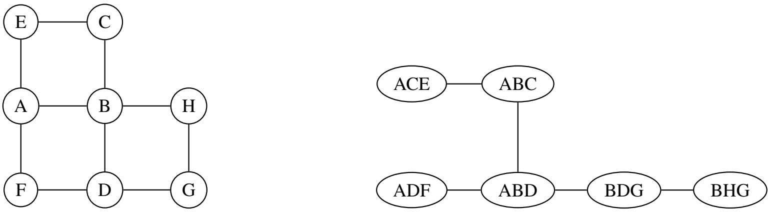


Figure 1. Interaction graph of a graphical model with 8 variables (one per node) and arity-2 cost functions (one per edge) (left) and one of its possible tree-decompositions (right).

$$F(X) = \sum_{f_i \in F} f_S(S)$$

and the goal that we are considering here is to minimize it. In probabilistic problems (i.e, the objective function has a probabilistic interpretation) this task is called *most probable explanation*. In non-probabilistic problems it is usually referred as *Weighted CSP*.

Solving a Graphical model is an NP hard problem, which means that the existence of polynomial algorithms is unlikely to exist. The design of effective algorithms for this problem has attracted a lot of research interest in the last decade [3,4].

2.2. *Interaction Graph*

The term Graphical Model comes from the existence of an underlying graph that captures important structural properties. Given a Graphical Model $P = \{X, D, F\}$, its *Interaction graph*, noted $G_P = (V, E)$, is an undirected graph with vertices V and edges E. There is one vertex $i \in V$ associated to each variable $x_i \in X$, and there is an edge $(i, j) \in E$ if and only if there some cost function $f_S \in F$ with $\{i, j\} \subseteq S$. Thus, the interaction graph tells pairs of variables that are linked (or connected) via cost functions. Figure 1 (left) shows the interaction graph of a graphical model with 8 variables (one per node) and 10 arity-2 cost functions (one per edge).

It is well-known that graphical models whose interaction graph is acyclic can be solved efficiently. For graphical models with a cyclic interaction graph, we can identify its acyclic sub-components through a tree-decomposition.

2.3. *Tree Decomposition*

A **tree-decomposition** of a graphical model $P = \{X, D, F\}$ is a tree $T = (V, A)$. For every vertex $e \in V$ there is a cluster $C_e \subseteq X$. The set of clusters must cover all the variables (i.e, $\cup_{e \in V} C_e = X$) and all the cost functions (for every $f_S \in F$ there is some cluster C_e such that $S \subseteq C_e$). Furthermore, if a variable x_i appears in two clusters C_e and C_k, it must also appear in all the clusters on the unique path from e to k (this is called the running intersection property).

The **tree-width** of a tree decomposition, noted w, is $\max_{e \in V}\{|C_e|\} - 1$. The tree-width of G is the minimum tree-width of all tree decompositions of G.

Figure 1 (right) shows a width 2 tree-decomposition of the graphical model whose cyclic interaction graph is depicted on the left.

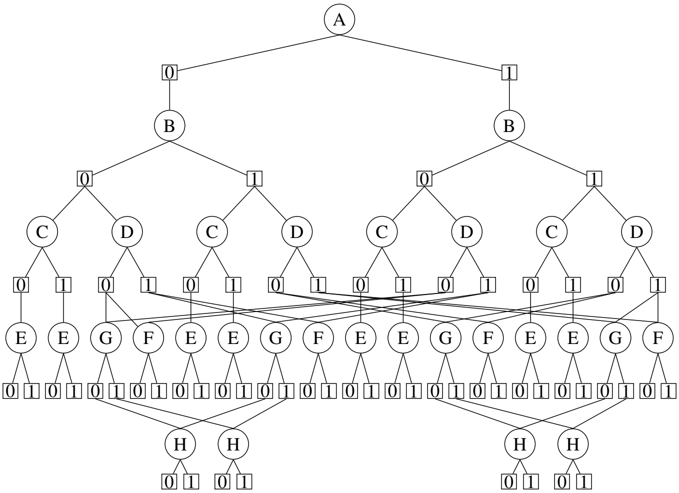


Figure 2. AND/OR graph search space.

2.4. Decomposition-based Backtracking Algorithms

Let $T = (V,A)$ be a tree decomposition of graphical model P. If we chose a node $r \in V$ as the tree root, then each tree node $e \in V$ has a parent $pa(e)$ and a set of children $ch(e)$. The separator of C_e is the set of common variables with its parent $S_e = C_e \cap pa(C_e)$. The set of proper variables of C_e is $V_e = C_e - S_e$. Note that proper variables of clusters define a partition of the variables. We denote $[i]$ the index of the cluster where x_i is a proper variable. Each cost function f_S is associated to the higher cluster that contains S and we denote $[S]$ the index of such cluster.

Decomposition-based backtracking algorithms assign variables in a top-down order with respect the decomposition. In other words, if variable x_i is a proper variable of cluster C_e (that is $[i] = e$), it cannot be assigned until all the variables of its parent $pa(C_e)$ have been assigned. Therefore, we can think of a partial assignment t as tree-structured. Corresponding to a top-down partial labeling of the variables in the decomposition.

We denote P_e the problem with all the variables of C_e and its descendent clusters and all the cost functions having in their domain at least one proper variable of these clusters. Consider a partial assignment t of the variables in P that includes the variables in C_e but does not include any proper variable of any child of C_e. Then for every C_k child of C_e, we denote $P_k(t)$ to P_k conditioned by t. Each one of these conditioned subproblems can be solved independently. Consequently, the search space traversed by this algorithms is the space of tree-structured partial assignments. To fully mimic the dynamic programming approach and inherit its good worst-case complexity, these algorithms may record the optimum of each solved subproblem, so they need not to solve it more than once. In terms of search space, this corresponds to merging nodes that correspond to identical subproblems. Figures 2 and 3 show two different search spaces for the tree decomposition of Figure 1 assuming binary domains. The two search spaces correspond to choosing two different roots for the tree-decomposition. Choosing the root producing the most cost-effective search space is the topic addressed in this paper.

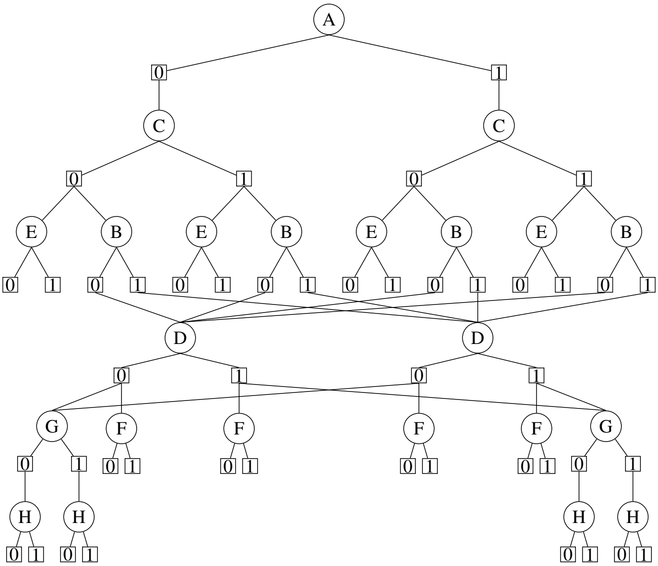


Figure 3. Another AND/OR graph search space for the same problem.

3. Experimental Design

In order to conduct our study we needed a diverse set of instances that were neither too easy (in order to make comparisons meaningful) nor too hard (so that executions would not exceed our computation budget). For that purpose we used two repositories:

- The evalgram repository (http://genoweb.toulouse.inra.fr/ degivry/evalgm/)
- The Cost Function Library (https://forgemia.inra.fr/thomas.schiex/cost-function-library/-/tree/master/)

Together they contain more than 16500 instances coming from different application domains such as *Bioinformatics*, *Boolean discrete* problems, B*ayesian netorkws*, *Airplane landing*, *Satelite observations*, *Graph coloring*, *Tractability-preserving Transformations of Global Cost Functions*, *Warehouse location problems* and many others.

We made a massive execution of all the instances using *toulbar2* default and selected instances whose execution time range from 10 to 30 minutes [3]. For each problem having selected instances we selected two of them. Then we computed their tree decomposition using the *minfill* heuristic [14] and eliminated hard instances having more than 100 clusters and instances having large tree widths (not suitable for decomposition-based algorithms). We further eliminated instances that were problematic for different reasons such as having several connected components or having soft global constraints. As a result we obtained 19 diverse and challenging instances.

[3] For obvious reasons (*i.e*, $16500 \times 0.5 = 344$ days-cpu), in this experiment we used all the machines that we had available. Although they were roughly similar, they were not identical, so cpu times between instances are not comparable

Instance Name	var.	dom.	const.	arity	costs	optimum	width	separ.	clust.
autocorr_bern50-13	50	5	4120	4	4171	747152	12	12	38
carseqtern_13_37	285	2 − 13	1046	3	1332	57	13	13	272
graph06_r	198	6 − 44	841	2	1040	4123	58	50	136
scen06-18reduc	82	4 − 26	327	2	409	3263	11	8	44
composed	83	8 − 10	624	2	707	2	46	46	34
rus_50_100_3_2	135	2	3407	2	3543	1340579	34	34	101
hole10	110	2	561	10	671	1	71	52	19
geo	50	20	466	2	516	1	22	22	24
sanr400_0.5.clq	400	2	39816	2	40217	387	381	381	19
aim	200	2	427	3	628	105	68	63	122
packupweighted1	707	2	2659	9	3367	186592	69	55	388
packupweighted2	613	2	2463	9	3077	93956	51	48	343
wellparhardtern9_9	163	1 − 3	1192	3	1355	44	61	59	102
wellpartern5_79	254	2 − 13	1046	3	1332	4035	22	16	197
parity_learn_48_24_6.2	459	2 − 19	4120	4	4171	5	23	19	436
or_chain_244.fg	750	2	1604	3	2355	397147782	84	64	591
cnf2.80.1000.266842	80	2	802	2	883	146	55	54	25
max_cut_50_500_1.asc	50	2	500	2	550	188	36	36	14
cnf3	150	2	710	3	861	64	101	98	48

Table 1. Benchmark Description. Composed, geo, aim, packupweighted1, packupweighted2 and cnf3 full names are composed-75-1-2-9_ext_100, geo50-20-d4-75-86_ext_1000, aim-200-1_6-yes1-4.cnf.mo, 978532fa-c730-11df-b070-00163e3d3b7c_l1, e69a0e36-9ef1-11df-9d4a-00163e46d37a_l1, _weightedregular, cnf2.80.1000.266842, respectively.

Table 1 contains information about the instances. It can be seen that the guarantee of benchmark diversity does not come only from the different origin of instance, but also from their syntactical structure. The number of variables (column 2) ranges from 50 to 750, the domain size (column 3) ranges from 2 to 44, the number of cost functions (column 4), the maximum arity of cost functions (column 5) ranges from 2 to 10. The number of different costs appearing over the cost functions (column 6) ranges from 409 to 40217. The optimal value (column 7) ranges from 1 to 397147782. In terms of decomposability, using the minfill heuristic the tree-width (column 8) ranges from 11 to 381 and the separator size (column 9) from 8 to 381. Finally, the number of clusters (column 10) ranges from 14 to 591.

Toulbar2 makes a sophisticated pre-process before starting the solving process. This pre-process may change slightly the problem structure (e.g. eliminating variables) and in turn the tree decomposition. To avoid that these changes could obfuscate our results, we transformed each instance to its VAC (that is, *virtual arc-consistent* [5]) pre-processed version (e.g. `"toulbar2 instance.wcsp -A -z=2"`) and computed a minfill tree decomposition again (e.g. `"toulbar2 instanceVAC.wcsp -B=1 -hbfs: -O=-3 -Z=1"`).

Then we solved each instance *instanceVAC.wcsp* with each node *r* of the tree decomposition as root using the BTD algorith [10] (e.g.`"toulbar2 instanceVAC.wcsp -O=-3 -B=1 -hbfs: -R=r"`). In order to make the comparison of cpu time meaningful in this experiment all executions were done using identical machines (*Fujitsu Primergy CX250 S1 with Intel Xeon E5-2660 @ 2,2 Ghz and 128G of RAM*). Jobs were managed

Instance Name	worst	mean	best	worst / best	mean / best
autocorr_bern50-13	3059	1479	669	4.57	2.21
carseqtern_13_37	*(17)	1849	498	211.76	108.8
graph06_r	*(106)	2861	0	5077.57	4035.69
scen06-18reduc	*(39)	3340	24	148.95	138.21
composed	*(19)	2072	0	50000	28780.4
rus_50_100_3_2	630	504	375	1.68	1.34
hole10	*(15)	3468	1417	2.54	2.32
geo	*(10)	2374	720	4.99	3.16
sanr400_0.5.clq	746	627	534	1.4	1.17
aim	*(117)	3513	329	30.77	29.79
packupweighted1	*(242)	2722	327	14.88	11.22
packupweighted2	2417	1033	134	18.01	7.7
wellparhardtern9_9	*(23)	1596	157	156.52	68.76
wellpartern5_79	*(17)	1031	92	211.76	60.35
parity_learn_48_24_6.2	1874	1132	491	3.82	2.3
or_chain_244.fg	43	6	0	60.37	9.01
cnf2.80.1000.266842	*(20)	3356	1860	180	161.17
max_cut_50_500_1.asc	*(1)	2431	1325	3600	2257.49
cfn3	*(27)	2614	235	133.33	94.84

Table 2. Results solving each instance with every tree decomposition cluster as root. Times (**worst**, **mean** and **best** columns) are given in seconds.

with SLURM queue system with each job requesting exclusive use of 1 core and 8Gb of RAM. To bound the duration of the experiment a time out was set to 3600 seconds [4]

Results from this experiment are reported in Table 2. The second and fourth columns report execution times for the worst and best roots. In the second column, an asterisk indicates that the time out has been reached as many times as the number within parenthesis. The third column reports the average time over all the roots or, equivalently, the expected time by choosing a root randomly. However, note that in most of the instances the average is underestimated, since the timeout is often reached for several roots. Columns 5 and 6 report ratios. Note again that ratios are underestimations in all the instances where the timeout is reached.

The main observation from the ratios is that most of the times choosing the right root has a dramatic effect in the algorithm's performance. In a couple of instances, the best root makes the algorithm thousands of times faster than the worst root and even than the expected time from a random root selection. In most instances the best root makes the algorithm around an order of magnitude faster. There is only one instance where the best root is more than half the time of the worst.

[4] Note that if every execution reached the timeout a total of 2954 hours (123 days) of cpu would be required

Instance Name	$S(T,\cdot)$	$Sd(T,\cdot)$	$CD(T,\cdot)$	$H(T,\cdot)$
autocorr_bern50-13	-	-	0.95	0.95
carseqtern_13_37	0.24	0.25	0.12	0.12
graph06_r	0.35	0.36	−0.29	−0.25
scen06-18reduc	−0.8	−0.79	−0.75	−0.84
composed	0.98	0.98	−0.89	−0.89
rus_50_100_3_2	0.07	.0.07	−0.02	−0.02
hole10	−0.86	−0.86	0.99	0.99
geo	−0.72	−0.72	0.71	0.62
sanr400_0.5.clq	−0.77	−0.77	0.74	0.61
aim	0.7	0.7	−0.27	0.38
packupweighted1	-	-	0.31	0.21
packupweighted2	-	-	-	−0.34
wellparhardtern9_9	−0.15	−0.15	−0.47	−0.45
wellpartern5_79	0.04	0.03	0.11	0.05
parity_learn_48_24_6.2	−0.06	−0.05	0.05	0.08
or_chain_244.fg	0.15	0.15	−0.48	−0.08
cnf2.80.1000.266842	0.7	0.71	−0.81	−0.8
max_cut_50_500_1.asc	−0.9	−0.9	0.74	0.74
cfn3	−0.63	−0.63	0.31	0.38

Table 3. Results solving each instance with every tree decomposition cluster as root. Correlation with four different measures.

The previous results confirm our initial hypothesis of the interest of finding automatic mechanisms to identify near optimal roots. Aiming at that, we looked for a simple predictor. In particular we considered four options:

- **Cluster Size:** The size of cluster $e \in V$, noted $S(T,e)$, is the number of variables that it contains $S(T,e) = |C_e|$. It seems reasonable choosing as root r the node with the largest size $S(T,r)$, because it means that the solving process will start with a large cluster of highly connected variables. This idea follows the well-known fail-fist principle which, in our context, means that the propagation effect of local consistency properties will become apparent as early as possible. In other words, selecting the node with the largest cluster represents a greedy way to produce good lower bounds quickly which in turn will produce good pruning.
- **Domain Aware Cluster Size:** The previous definition takes all the variables as equivalent. If we want to use cluster size as a proxy of search space size a better approach is to take into account domain sizes. Accordingly, we define,

$$Sd(T,e) = \sum_{x_i \in C_e} \log |d_i|$$

  It seems reasonable choosing the root r with the largest $Sd(T,r)$ as a refinement of cluster size. Note that for instances in which all the variables have the same domain size, $Sd(T,r)$ and $S(T,r)$ are equivalent.
- **Cluster Decomposition Size:** The cluster decomposition size with root $r \in V$, noted $CD(T,r)$, is the size of the largest sub-problem after the assignment of the

variables in C_e. Formally, $CD(T,r)$ is the maximum $|D(T,e)|$ over all $e \in ch(r)$ defined as,

$$D(T,e) = V_e \cup \cup_{k \in ch(e)} D(T,k)$$

where $ch(e)$ is implicit by the choice of r as the root. It seems reasonable choosing as root the node with the smallest decomposition because it is the root that breaks the problem into sub-problems whose largest one is minimal. In other words, the assignment of its variables produces the minimal largest sub-problem. Note that this measure also favors rooting with large clusters since a large root leaves less variables for the sub-problems

- **Cluster Height:** The previous definition only considers the decomposition after the root assignment. It does not account for subsequent deeper decompositions. To incorporate that we define the height of a tree decomposition rooted with $r \in V$ as the size of the longest path starting from r. Formally, the height of root r is $H(T,r)$ with,

$$H(T,e) = |V_e| + \max_{k \in ch(e} H(T,k)$$

where $ch(e)$ is implicit by the choice of r as the root. It seems reasonable choosing as root the node with the lowest height, $\min_r H(T,r)$ because it means that backtracking will search on tree-structured assignments of minimum height (i.e, they will be the widest and most shallow).

Table 3 reports, for each instance and its given tree-decomposition T, the correlation between cpu time and the four measurements: cluster size $S(T,\cdot)$, domain-aware cluster size $Sd(T,\cdot)$, cluster decomposition size $CD(T,\cdot)$ and cluster height $H(T,\cdot)$. In the *autocorr* and *packupweighted* instances some measurements did not change over roots, so correlation could not be computed. We were expecting to find negative correlations between cpu and (domain aware) cluster size, and positive correlations between cpu and decomposition and height. Some instances (e.g. hole10, geo, max-cut, cnf) behaved as we expected and the correlations were high, which means that our four measures predict the quality of the root. In some other instances (e.g. rus, parity,...) correlations are very low, so none of our measures do not capture the quality of the root. Surprisingly, in some other instances (e.g. graph, composed,...) correlations were high but with the sign opposed to our conjecture, so our measures worked totally counter intuitively. In summary, simple synctactic measurements of the tree decomposition do not seem to capture the quality of the clusters as roots.

4. Conclusions and Future Work

In this paper we report our preliminary results in the quest of identifying near optimal roots of tree decompositions to be used in decomposition-based backtracking algorithms for Graphical Models. We proposed 4 different simple criteria based on synctactical measures of the tree-decomposition.

We created a small benchmark selecting from two well-known repositories 19 highly diverse challenging instances. We performed a systematic experiment solving each instance with every root and checked for correlations. The first lesson to be extracted from the experiment is the confirmation of the impact of the root on the algorithm performance, which validates the importance of our work. The second lesson is about the significance of the proposed measures: *i*) none of them showed significant correlation with all the instances, and *ii*) some instances correlated as expected, but others correlated counter intuitively. Therefore, we conclude that although choosing the root is important, simple measures based on syntactical features of the tree decomposition do not seem to predict well good clusters. Accordingly, in our future work we will search for more informed data (e.g. taking into account cost functions) and more sophisticated techniques to combine it (e.g. machine learning).

References

[1] Bodlaender HL, Grigoriev A, Koster AMCA. Treewidth Lower Bounds with Brambles. Algorithmica. 2008;51(1):81-98.

[2] Robertson N, Seymour PD. Graph minors. III. Planar tree-width. J Comb Theory, Ser B. 1984;36(1):49-64.

[3] Dechter R. Reasoning with Probabilistic and Deterministic Graphical Models: Exact Algorithms, Second Edition. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers; 2019.

[4] Darwiche A. Modeling and Reasoning with Bayesian Networks. Cambridge University Press; 2009.

[5] Cooper MC, de Givry S, Sánchez-Fibla M, Schiex T, Zytnicki M, Werner T. Soft arc consistency revisited. Artif Intell. 2010;174(7-8):449-78.

[6] Dechter R. Bucket Elimination: A Unifying Framework for Reasoning. Artif Intell. 1999;113(1-2):41-85.

[7] Bertelè U, Brioschi F. On Non-serial Dynamic Programming. J Comb Theory, Ser A. 1973;14(2):137-48.

[8] Cooper MC, de Givry S, Schiex T. Graphical Models: Queries, Complexity, Algorithms (Tutorial). In: Paul C, Bläser M, editors. 37th International Symposium on Theoretical Aspects of Computer Science, STACS 2020, March 10-13, 2020, Montpellier, France. vol. 154 of LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik; 2020. p. 4:1-4:22.

[9] Nils N. Artificial Intelligence: A New Synthesis. Morgan Kaufmann; 1998.

[10] Terrioux C, Jégou P. Bounded Backtracking for the Valued Constraint Satisfaction Problems. In: Rossi F, editor. Principles and Practice of Constraint Programming - CP 2003, 9th International Conference, CP 2003, Kinsale, Ireland, September 29 - October 3, 2003, Proceedings. vol. 2833 of Lecture Notes in Computer Science. Springer; 2003. p. 709-23.

[11] Marinescu R, Dechter R. AND/OR Branch-and-Bound for Graphical Models. In: Kaelbling LP, Saffiotti A, editors. IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence, Edinburgh, Scotland, UK, July 30 - August 5, 2005. Professional Book Center; 2005. p. 224-9.

[12] Allouche D, de Givry S, Katsirelos G, Schiex T, Zytnicki M. Anytime Hybrid Best-First Search with Tree Decomposition for Weighted CSP. In: Pesant G, editor. Principles and Practice of Constraint Programming - 21st International Conference, CP 2015, Cork, Ireland, August 31 - September 4, 2015, Proceedings. vol. 9255 of Lecture Notes in Computer Science. Springer; 2015. p. 12-29.

[13] Jégou P, Terrioux C. Combining restarts, nogoods and bag-connected decompositions for solving CSPs. Constraints An Int J. 2017;22(2):191-229.

[14] Dechter R. Constraint processing. Elsevier Morgan Kaufmann; 2003.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210117

Exploring Lifted Planning Encodings in Essence Prime

Joan ESPASA, [a] Jordi COLL, [b] Ian MIGUEL, [a] and Mateu VILLARET [c]

[a] *School of Computer Science, University of St Andrews, St Andrews KY16 9SX, UK*
e-mail: {jea20,ijm}@st-andrews.ac.uk

[b] *Aix Marseille Univ, Université de Toulon, CNRS, LIS, Marseille, France*
e-mail: jordi.coll@lis-lab.fr

[c] *Departament d'Informàtica, Matemàtica Aplicada i Estadística*
Universitat de Girona, E-17003 Girona, Spain
e-mail: mateu.villaret@udg.edu

Abstract. State-space planning is the *de-facto* search method of the automated planning community. Planning problems are typically expressed in the Planning Domain Definition Language (PDDL), where action and variable templates describe the sets of actions and variables that occur in the problem. Typically, a planner begins by generating the full set of instantiations of these templates, which in turn are used to derive useful heuristics that guide the search. Thanks to this success, there has been limited research in other directions.

We explore a different approach, keeping the compact representation by directly reformulating the problem in PDDL into ESSENCE PRIME, a Constraint Programming language with support for distinct solving technologies including SAT and SMT. In particular, we explore two different encodings from PDDL to ESSENCE PRIME, how they represent action parameters, and their performance. The encodings are able to maintain the compactness of the PDDL representation, and while they differ slightly, they perform quite differently on various instances from the International Planning Competition.

Keywords. Automated Planning, Constraint Programming, Modelling, Reformulation

1. Introduction

Given a model of the environment, a planning problem asks us to find a sequence of actions that leads from an initial state to a given goal state. These models are typically expressed in the Planning Domain Definition Language [1] (PDDL). The user describes the problem in terms of predicates, actions and functions with parameters. In turn, these parameters can be instantiated with a set of defined objects.

A simple example of a planning problem is a logistics problem, expressed in PDDL in Figure 1, where we must transport Bob and Alice from the city of Barcelona to the airport, so they are finally able to embark on a plane that will take them home. The initial state is that both Alice and Bob are at Barcelona, the goal is having them embarked in a plane at the Airport, and the possible actions are `move` (change person of location) and `embark` (person into plane at a location).

In spite of having a compact model representation, having read the model, a planner typically begins by producing a totally *grounded* representation. The action of grounding (or instantiation) will replace all variables that represent parameters in actions by their possible values, creating all the possible instantiations of the actions. After grounding, no variables are left free and all valid instantiations of predicates and functions in the actions are computed. The size of the fully grounded planning problem is exponential in the maximum number of arguments of all the actions.

Depending on the original problem and how the task is grounded, this growth can result in an instance that cannot be efficiently handled. There have been approaches that try to alleviate this grounding problem in various ways. For example, one could ground only relevant parts of the problem [2,3], make clever representations of actions [4] or simplify the input problem [5]. The opposite of grounded planning would be *lifted* planning, where grounding is fully avoided. Grounding is normally seen as a necessary step, and there are very few approaches to lifted planning that skip grounding entirely [6,7,8]. These approaches are not that popular mainly due to the efficiency of computing informative heuristics on a grounded representation, which are difficult to compute at the lifted level. Also, reasoning at a more abstract level is typically more difficult.

Herein we try to avoid grounding as much as possible by using the expressivity of ESSENCE PRIME [9], a declarative constraint modelling language. Moreover, we take advantage of SAVILE ROW [10], a sophisticated constraint reformulation tool supporting ESSENCE PRIME that is able to generate SAT and SMT [11], or CSP [12] instances.

Our contributions in this paper are two different lifted encodings from PDDL to ESSENCE PRIME, which differ in how they represent action parameters. The encodings maintain the compactness of the PDDL representation, and while they differ slightly, they perform quite differently. We also report results with backend solvers for SAT, SMT and for a lazy clause generation [13] (LCG) constraint solver. The rest of the paper proceeds as follows. In Section 2 we recall the theoretical framework. In Section 3 we propose the encodings from PDDL to ESSENCE PRIME. Section 4 is devoted to experimental evaluation of the encodings. Finally, Section 5 discusses future work and concludes.

2. Preliminaries on Automated Planning

This paper considers numeric planning problems, which extend propositional planning with numeric state variables. We formally define the numeric planning problem only in terms of the grounded representation of the problem.

Definition 1 (Numeric Planning Problem)**.** A planning problem can be defined as a tuple $\prod = \langle B, O, F, X, A, I, G \rangle$, where: B is a set of names for all the objects, O is a set of object state variables, F is a set of propositional state variables, X is a set of numeric state variables, A is a set of actions, I is the initial state and G is the goal.

An action $a \in A$ is defined as a tuple $a = \langle Pre_a, Eff_a \rangle$, where Pre_a refers to the precondition and Eff_a to the effects of the action.

Definition 2 (State)**.** Given a planning problem $\prod$, a *state* is a variable-assignment (or valuation) function over state variables $O \cup F \cup X$, which maps each $o \in O$ to an object in B, each $f \in F$ into a truth value, and each $x \in X$ to an integer. A state is represented a set of ordered pairs $\{(v_1, z_1), (v_2, z_2), \ldots, (v_n, z_n)\}$, where each v_i is the variable and z_i the value mapped to it.

An *object condition* has the form $\zeta \otimes b$, where ζ is an expression over O, $\otimes \in \{=, \neq\}$ and b is an object in B. A *numeric condition* has the form $\zeta \otimes k$, where ζ is a linear integer arithmetic expression over X, $\otimes \in \{\leq, <, =, >, \geq\}$ and k is an integer constant.

Preconditions (*Pre*) and the goal G are sets (conjunctions) of object conditions, numeric conditions and propositions. Action effects (*Eff*) are sets of assignments to propositional variables, assignments to object variables and increase/decrease/assign a numeric variable by a numeric expression. A *conditional effect* is a pair $\langle c, e \rangle$ where c is a set of object, numeric and propositional conditions; and e is an effect. e is applied only if c is satisfied in the state where the action is applied.

An action a is *applicable* in a state s only if its preconditions are satisfied in s ($s \models Pre_a$) and the applied numeric, object and propositional effects do not induce conflicting assignments. The outcome after the application of an action a will be the state where variables that are assigned in Eff_a take their new value, and variables not referenced in Eff_a keep their current values.

A sequence of actions $\langle a_0, \ldots, a_{n-1} \rangle$ is called a *plan*. We say that the application of a plan starting from the initial state I brings the system to a state s_n. If each action is applicable in the state resulting from the application of the previous action and the final state satisfies the goal (i.e., $s_n \models G$), the sequence of actions is a a *valid plan*. A planning problem has a solution if a valid plan can be found for the problem.

3. Encodings

In this section we propose various encodings for a numeric planning problem. First we will explain how the planning as satisfiability approach works, then in what kind of input we will receive the planning problem and finally the proposed encodings.

3.1. Planning as Satisfiability

As is typical in the planning as SAT or as CSP approaches [14,15,16], we will solve the planning problem by considering a sequence of CSPs ϕ_0, ϕ_1, ϕ_2, ..., where ϕ_i encodes the existence of a plan that allows to reach a goal state from the initial state in i steps. The solving procedure will test the satisfiability of ϕ_0, ϕ_1, ϕ_2, and so on, until a satisfiable formula ϕ_n is found, proving the existence of a valid plan of n steps.

Each ϕ formula will need variables to represent the state for each step and need to define the values of the variables in the initial step. Then, it will also need some variables to represent which action is executed at each step. We will need to make sure that if an action is executed, its precondition holds with respect to the problem variables. We will need to make sure that the goal conditions are met and we will do it by adding some constraints on the variables representing the state of the final step. Finally, we will need to make *frame axioms* explicit, i.e. constraints that specify that if no action has modified a variable, it keeps its value between steps. Semantics such as the $\forall$ or $\exists$-step [17] allow parallel actions, but here just one action will be executed per time step.

3.2. Planning Domain Definition Language (PDDL)

In contrast to the formal definition of a planning problem given in Section 2, PDDL allows the specification of problems in a lifted manner. Although being normally represented this way, most solving approaches ground the problems.

```
(define (domain transport)
 (:types person aircraft - locatable
        location - object)
 (:predicates (at ?p - locatable ?l - location)
              (in ?p - person ?a - aircraft))
 (:functions (seats ?p - aircraft) - number)
 (:action move
    :parameters (?p - person ?from ?to - location)
    :precondition (at ?p ?from)
    :effect (and (not (at ?p ?from)) (at ?p ?to)))
 (:action embark
    :parameters (?p - person ?l - location ?a - aircraft)
    :precondition (and (at ?p ?l) (at ?a ?l) (> (seats ?a) 0))
    :effect (and (not (at ?p ?l)) (in ?p ?a) (decrease (seats ?a) 1))))

(define (problem example)
 (:domain transport)
 (:objects plane - aircraft
           Bob Alice - person
           Barcelona Airport - location)
 (:init (at Bob Barcelona) (at Alice Barcelona)
        (at plane Airport) (= (seats plane) 2))
 (:goal (and (in Bob plane) (in Alice plane))))
```

Figure 1. Domain and problem file in PDDL, representing the problem of moving Bob and Alice from Barcelona to a plane in the airport. A valid plan for the problem would be: `(move Bob Barcelona Airport)`, `(move Alice Barcelona Airport)`, `(embark Bob Airport plane)` and `(embark Alice Airport plane)`.

A *fluent*, in the area of automated planning, refers to a variable that represents some attribute of the problem and changes over time. Roughly speaking, our framework will be numeric planning. More concretely, our formalism will derive from PDDL 2.1 [18], without temporal semantics or metric optimizations. We also consider functional strips semantics [19], incorporated in the recent revisions of the PDDL. This means that, apart from reasoning with integer fluents, we will be able to have actions that work with objects and refer to attributes of these objects. Therefore, a fluent declared as `(location ?p - object) - place` will be able to express where objects are, and expressions like `(= (location plane) (location person))` or `(> (fuel plane) 10)` will be valid.

Even though planning formalisms do not consider templates, they are widely used in PDDL to make the representation compact. Types are also used in PDDL to make the problem more readable and to give more information to the planners. It can be seen in Figure 1 how types, templates for actions, predicates and functions are expressed. As our input will be a problem defined in the PDDL language, we will need to directly consider them. In fact, the instantiations of the predicate templates will correspond to the predicate state variables of the planning problem at hand, and the instantiations of the function templates will correspond to the object and numeric state variables of the planning problem, depending on its return type.

Templates can be *state variable templates* or *action templates*. These are comprised of a name and a sequence of typed parameters, or "ordinary" variables. For example, consider `(location ?p - object) - place`, being an object state variable template. Its name is `location` and its parameters, the sequence [`?p`], where the only parameter

`?p` has the name `p` and the `object` type. The domain of this object state variable is the set of objects with type `place` in the problem.

For instance, in the PDDL specification, expressions such as preconditions and effects can also contain variables, belonging to the action template parameters. For example, the effect `(and (not (at ?p ?from)) (at ?p ?to))` belonging to the `move` action template in Figure 1 contains three variables: `p`, `from` and `to`.

3.3. Basic Encoding

In this section we describe formulas ϕ_h, that is, the existence of a valid plan with h actions. Again, our purpose in this work is to encode PDDL instances into ESSENCE PRIME in a lifted manner. Roughly, a grounded representation would have a Boolean variable stating whether action $move\_alice\_Barcelona\_airport^t$ is performed in a given time step t. Instead, for each time step t, we will have an integer variable stating which action template is applied and an integer variable per parameter of each action template stating what particular object is used as parameter of that action. Moreover, and for each time step t, we will also have a Boolean or Integer ESSENCE PRIME variable (CP variable) for each concrete instantiation of each state variable template.

To express the encoding, we will need some auxiliary definitions. Let E be the set of types specified in the PDDL model. Each object $b \in B$ has a type associated with it. Also, each type $e \in E$ has a domain associated to it, being $Domain_e \subseteq B$. Let A_T be the set of action templates in the PDDL problem. Similarly, O_T, F_T and X_T will be the sets of object, propositional and numeric state variable templates, respectively.

Let V be the set $O \cup F \cup X$, representing the set of all state variables, without taking their type into account. V_T will represent the set of all state variable templates. For $x \in A_T \cup V_T$, let $Parameters_x$ be the sequence $[z_1, \dots, z_n]$, representing the parameters of the template. For each parameter z_i, let $Type_{z_i}$ be the type associated to z_i, and $Name_{z_i}$ its name. Let $l(k) \rightarrow \mathbb{Z}$ be an injective function defined for all $k \in B \cup A$. It serves as a labelling function, that maps an object or action to a unique integer. This will be useful to later encode objects and object state variables as integers and integer state variables respectively. We will start by introducing the following CP variables:

$$state_v^t \qquad \forall v \in V, \forall t \in 0..h \tag{1}$$

$$action^t \qquad \forall t \in 1..h \tag{2}$$

$$param_{a,i}^t \qquad \forall a \in A_T, \forall i \in Parameters_a, \forall t \in 1..h \tag{3}$$

Variables $state_v^t$ hold the value of state variable v in step t. This representation corresponds to a new CP variable for each grounded state variable. Variables $action^t$ express which action is scheduled at time step t. The domain of these $action^t$ variables is $\{l(a) \mid a \in A_T\}$, being the set of integers the labelling function l assigns to the problem action templates. Finally, variables $param_{a,i}^t$ denote the value of i-th parameter in action template a at each step t. Each of these variables will have a domain of the parameter type. Note that variables introduced in (2) and (3) correspond to the action templates. With this representation there is no need to ground all the possible instantiations of the actions, and the solver will be responsible for choosing which action template is executed and with which parameters. We state initial and goal states:

$$state_v^0 = z \qquad \forall (v,z) \in I \tag{4}$$

$$g^h \qquad \forall g \in G \tag{5}$$

where G is a conjunction of conditions on state variables, and g^h is the ESSENCE PRIME translation of these conditions on CP variables $state_v^h$ for all variables v in the conditions of G. Note that the initial state must be fully specified.

Frame axioms express that, if a given state variable has changed from one time step to the next, it is because an action that is able to change it has been executed.

$$state_v^{t-1} \neq state_v^t \rightarrow \bigvee_{\substack{\forall a \in A_T,\\ \forall m \in modify(a,v)}} \left(action^t = l(a) \wedge \bigwedge_{\forall (j,o) \in m} param_{a,j}^t = l(o) \right) \quad \begin{array}{l} \forall t \in 1..h, \\ \forall v \in V \end{array} \tag{6}$$

Given an action template a and a state variable v, the function $modify(a,v)$ returns the set of all combinations of parameter assignments (expressed as a pair (j,o)) that make action a modify variable v. For instance, the state variable `at(Bob,Barcelona)` is modified by action template `move`, with the following set of parameter assignments:

$\{\{(\texttt{p},\texttt{Bob}),(\texttt{from},\texttt{Barcelona}),(\texttt{to},\texttt{airport})\},\ \{(\texttt{p},\texttt{Bob}),(\texttt{from},\texttt{airport}),(\texttt{to},\texttt{Barcelona})\},...\}$

Finally, actions are expressed

$$action^t = l(a) \rightarrow Pre_a^t \wedge Eff_a^t \qquad \forall a \in A_T, \forall t \in 1..h \tag{7}$$

Preconditions are sets of conditions and effects are sets of assignments. When translating Pre_a^t and Eff_a^t into ESSENCE PRIME, we use the *element* global constraint to access the corresponding state variables according to the values given to the action parameters. The translation of conditions and state variable assignments to ESSENCE PRIME is straightforward. However, conditions and right hand sides of assignments will consult the state variables of time $t-1$, and left hand side of the assignments will update state variables of time t. For instance, when considering the effect on the number of free seats in the `embark` action: `seats[embark_a[k],k] = seats[embark_a[k],k-1]-1`.

3.4. Encoding Compaction

Approximations such as the ∀-step or ∃-step semantics [17] allow parallel actions as long as they are not interfering. For now our encoding assumes that one action will be executed per time step. With one action executed per time step, we can see that most of the variables from (3) are rarely used. That is, only the parameters belonging to the selected action are used, and the others are ignored. Here we introduce two variants of the encoding with the aim of reducing the total number of variables: *Type sharing* and *Max Parameters*. They differ in how parameters are treated, as shown in Figure 2.

Before explaining the encodings, we introduce the concept of a *substitution* (or *renaming*) σ: a partial mapping from variables to variables. It can be represented as a function by a set of bindings of variables to variables. That is, if $\sigma = \{x_1 \mapsto y_1, \ldots, x_n \mapsto y_n\}$, then $\sigma(x_i) = y_i$ for all i in $1..n$, and $\sigma(x) = x$ for every other variable. Using an infix

Original PDDL representation

```
fly(?p - plane ?from ?to - loc)      unload(?p - plane ?x - package)
load(?p - plane ?x - package)
```

Standard encoding	**Type Sharing**	**Max parameters**
`fly`$(p_{fly,1}, p_{fly,2}, p_{fly,3})$ `unload`$(p_{unload,1}, p_{unload,2})$ `load`$(p_{load,1}, p_{load,2})$	`fly`$(p_{plane,1}, p_{loc,1}, p_{loc,2})$ `unload`$(p_{plane,1}, p_{package,1})$ `load`$(p_{plane,1}, p_{package,1})$	`fly`(p_1, p_2, p_3) `unload`(p_1, p_2) `load`(p_1, p_2)
Set of parameters	**Set of parameters**	**Set of parameters**
$\{p_{fly,1}, p_{fly,2}, p_{fly,3}, p_{unload,1}, p_{unload,2}, p_{load,1}, p_{load,2}\}$	$\{p_{plane,1}, p_{package,1}, p_{loc,1}, p_{loc,2}\}$	$\{p_1, p_2, p_3\}$

Figure 2. Example of how parameters are shared in the various encodings for the planes domain. For each encoding (standard, type sharing, max parameters), the corresponding set of parameters is shown below.

notation and given any expression τ containing variables, $\tau\sigma$ is τ with all the contained variables replaced, as specified by σ. For example, given a substitution $\sigma = \{p \mapsto q\}$ and the term representing an effect $\tau =$ `(and (not (at ?p ?from)) (at ?p ?to))`, the result of $\tau\sigma$ would be `(and (not (at ?q ?from)) (at ?q ?to))`.

3.4.1. Type Sharing

Although actions can have many parameters, they typically have few parameters of the same type. Therefore, in this encoding each action parameter of a given type is replaced by a new parameter that is shared by all the actions that need a parameter of that type.

Let C_e for each $e \in E$ be the maximum number of parameters on all actions that share type e. Then, variables introduced in (3) are substituted with

$$param^t_{e,i} \qquad \forall e \in E, \forall i \in 1..C_e, \forall t \in 1..h \tag{8}$$

Example 1. If the PDDL action that has most parameters with the `place` type is an action such as `move(?p - person, ?from - place, ?to - place)`, then $C_{place} = 2$. Then, the previous Equation will introduce parameter variables $param^t_{person,1}$, $param^t_{place,1}$ and $param^t_{place,2}$ for each time step.

Given an action template $a \in A_T$, a parameter $q \in Parameters_a$ and its type $Type_q \in E$, let $pos(q,a) = [z \mid z \in Parameters_a, Type_z = Type_q]$. That is, the subsequence of parameters of a that have the same type as q.

Then, we can define a substitution σ_a for every action $a \in A_T$, such that

$$\sigma_a = \{param_{a,q} \mapsto param_{e,i} \mid q \in Parameters_a, e = Type_q, i \in 1..|pos(q,a)|, pos(q,a)[i] = q\} \tag{9}$$

Finally, to Equation (7) is modified to use these new parameter variables

$$action^t = l(a) \to Pre^t_a\sigma_a \wedge Eff^t_a\sigma_a \qquad \forall a \in A_T, \forall t \in 1..h \tag{10}$$

Following Example 1, this will substitute all appearances on the *Pre* and *Eff* of $?p$ by $param_{person,1}$ and so on.

	depots(3)		driverlog(10)		planes(8)		zenotravel(9)		total
RanTanPlan-SMT(LIA)	1	(4809.4)	7	(3174.8)	3	(4600.2)	8	(945.4)	19
SR-SAT T. Sharing	3	(1354.3)	**10**	**(53.6)**	8	(758.9)	4	(4049.6)	25
SR-SAT Max. Par	2	(2614.1)	10	(814.1)	0	(7200.0)	7	(2143.5)	19
SR-LCG T. Sharing	2	(3293.8)	7	(2631.1)	3	(4708.8)	7	(1633.2)	19
SR-LCG Max. Par	1	(5763.0)	0	(*)	0	(*)	4	(4087.1)	5
SR-SMT(BV) T. Sharing	**3**	**(889.9)**	10	(205.0)	**8**	**(462.2)**	**9**	**(142.8)**	30
SR-SMT(BV) Max. Par	0	**(*)**	0	(*)	0	(*)	0	(*)	0

Table 1. For each domain and configuration: left, number of solved instances; right, mean solving time in seconds, counting timeouts as 7200 seconds. We only consider the subset of instances solved by some setting, and the subset sizes are next to domain name. All instances for cells with (*) have run out of memory.

3.4.2. *Max parameters*

An alternative approach is to share parameters independently of their types. That is, instead of dedicated parameter variables for each action, we will only declare n parameters, where n is equal to the number of parameters of the action with most parameters. Formally, $n = max(\{|Parameters_a| \mid a \in A_T\})$. These parameters will be representing different types depending on which action is executed. Therefore, the domain of each one will be the union of all possible objects. We will again substitute variables in (3) by

$$param^t_q \qquad\qquad \forall q \in 1..n, \forall t \in 1..h \tag{11}$$

Now, let σ_a be a substitution for every action $a \in A_T$, such that $\sigma_a = \{param_{a,q} \mapsto param_q | q \in Parameters_a\}$. This substitution will replace the mentioned parameters in the action by the new declared parameters in (11). Finally, Equation (7) is also modified to use these new variables

$$action^t = l(a) \rightarrow Pre^t_a\sigma_a \wedge Eff^t_a\sigma_a \qquad\qquad \forall a \in A_T, \forall t \in 1..h \tag{12}$$

To improve the encoding, if using a CSP solver as a backend, a *table* constraint can be added to the ESSENCE PRIME model to limit the possible values of the parameters, depending on the action chosen. Hence, once an action has been decided, the domains of the parameters are restricted to its declared types.

4. Experimental Evaluation

In this section we evaluate the performance of the presented encodings by solving a set of numeric planning problems coming from the third IPC [20]. These domains contain integer numeric fluents without quantified preconditions, as the rest of the domains contain features that we still do not support. These domains are: *Zenotravel*, *Driverlog*, *Depots*. The *Planes* domain from [21] is also considered since it has an interesting numerical component. Although some domains give various optimization criteria, we only consider the problem of finding a valid plan minimizing the total makespan, i.e. number of steps. As noted, our approach reformulates the PDDL description to the ESSENCE PRIME language. In turn, this ESSENCE PRIME model is given as input to SAVILE ROW [10] to generate a SAT, SMT or CSP model. Finally we use Glucose [22], Chuffed[1] and Boolector [23] as the backend solvers. We validated the usefulness of the SAVILE ROW preprocessing steps suchs as common subexpression elimination [24,25] or symmetry breaking capabilities by turning them off and determining that solving times were significantly increased, at least by a factor of two. To compare the presented encodings with a fully grounded approach, we use the linear planning as SMT encoding provided by RanTanPlan [21]. The experiments were run on a AMD Opteron® 6272 Processor. Each process was given a limit of 4GB of memory and 1 hour timeout.

We do not consider the basic encoding without compacting action parameters, as it behaves worse than the two proposed improvements. Table 1 shows the number of solved instances and average solving time for each domain and each considered approach. We can observe that there are differences in the performance of the different approaches in the different domains, but the SR-SMT approach is generally better than the other ones.

The *Depots* domain seems too big, as all the approaches are only able to return a solution for a very few instances. If we look at the *Driverlog*, *Zenotravel* and *Planes* domains, the different approaches differ between them. The lifted approaches are generally better than RanTanPlan which uses grounding, but *Zenotravel* is harder for the lifted approaches except for *SR-SMT(BV)*. The Type Sharing encoding is better in general for all solving approaches and all considered problems. Even though the Max. Parameters encoding generates fewer parameters, it uses the maximum possible domain size for each parameter. This could imply that parameters with small domain are encoded using unnecessarily large domains, and could be the reason why we have many memory outs with Max. Par. We have observed that the action parameters of *Zenotravel* instances have relatively balanced domain sizes in comparison with the other domains, explaining why *SR-SAT Max. Par* does not behave worse than *SR-SAT T. Sharing* in *Zenotravel*.

Summing up, we observe that lifted encodings with T. Sharing are generally better than the grounded encoding, and in particular *SR-SMT(BV) T. Sharing* is the overall best approach. However, further experiments are needed to identify to what extent this is due to using a lifted encoding. Other aspects such as the used SMT theory or the reformulations performed by SAVILE ROW could also play an important role.

5. Conclusions and Future Work

We have presented two lifted approaches to encoding planning problems as CSP, and experimented with different solving back ends via SAVILE ROW. One configuration outperformed the fully grounded linear encoding of RanTanPlan, which also solves numeric planning as SMT. The relative performance of the two encodings depends on the number of action parameters and their domain sizes. In future work, a preprocess could select an encoding based on problem structure. The encodings could also be improved by considering the symmetries between successive application of different actions, or by incorporating the application of various actions in the same step.

Acknowledgements This work is supported by UK EPSRC EP/P015638/1 and EP/V027182/1, by the MICINN/FEDER, UE (RTI2018-095609-B-I00), by the French Agence Nationale de la Recherche, reference ANR-19-CHIA-0013-01, and by Archimedes institute, Aix-Marseille University.

[1] https://github.com/chuffed/chuffed

References

[1] Haslum P, Lipovetzky N, Magazzeni D, Muise C. An Introduction to the Planning Domain Definition Language. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers; 2019.

[2] Helmert M. Concise finite-domain representations for PDDL planning tasks. Artificial Intelligence. 2009;173(5-6):503–535.

[3] Gnad D, Torralba A, Domınguez M, Areces C, Bustos F. Learning How to Ground a Plan - Partial Grounding in Classical Planning. In: AAAI; 2019. p. 7602–7609.

[4] Areces C, Bustos F, Dominguez MA, Hoffmann J. Optimizing Planning Domains by Automatic Action Schema Splitting. In: ICAPS; 2014. p. 11–19.

[5] Masoumi A, Antoniazzi M, Soutchanski M. Modeling Organic Chemistry and Planning Organic Synthesis. In: GCAI; 2015. p. 176–195.

[6] Penberthy JS, Weld DS. UCPOP: A Sound, Complete, Partial Order Planner for ADL. In: KR'92; 1992. p. 103–114.

[7] Bofill M, Espasa J, Villaret M. Efficient SMT Encodings for the Petrobras Domain. In: ModRef. Lyon, France; 2014. p. 68–84.

[8] Correa AB, Pommerening F, Helmert M, Francès G. Lifted Successor Generation Using Query Optimization Techniques. In: ICAPS; 2020. p. 80–89.

[9] Nightingale P, Rendl A. Essence' Description. 2016;ArXiv:1601.02865 [cs.AI].

[10] Nightingale P, Akgün Ö, Gent IP, Jefferson C, Miguel I, Spracklen P. Automatically improving constraint models in Savile Row. Artificial Intelligence. 2017;251:35–61.

[11] Biere A, Heule M, van Maaren H, Walsh T. Handbook of Satisfiability. vol. 326. IOS press; 2021.

[12] Rossi F, Van Beek P, Walsh T. Handbook of constraint programming. Elsevier; 2006.

[13] Ohrimenko O, Stuckey PJ, Codish M. Propagation via lazy clause generation. Constraints An Int J. 2009;14(3):357–391.

[14] Kautz HA, Selman B. Planning as Satisfiability. In: ECAI; 1992. p. 359–363.

[15] van Beek P, Chen X. CPlan: A Constraint Programming Approach to Planning. In: Sixteenth National Conference on AI and Eleventh Conference on Innovative Applications of AI; 1999. p. 585–590.

[16] Miguel I, Jarvis P, Shen Q. Flexible graphplan. In: ECAI; 2000. p. 506–510.

[17] Rintanen J, Heljanko K, Niemelä I. Planning as Satisfiability: Parallel Plans and Algorithms for Plan Search. Artificial Intelligence. 2006;170(12-13):1031–1080.

[18] Fox M, Long D. PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains. Journal of Artificial Intelligence Research (JAIR). 2003;20:61–124.

[19] Geffner H. Functional STRIPS: a more flexible language for planning and problem solving. In: Logic-based artificial intelligence. Springer; 2000. p. 187–209.

[20] Long D, Fox M. The 3rd International Planning Competition: Results and Analysis. Journal of Artificial Intelligence Research (JAIR). 2003;20:1–59.

[21] Bofill M, Espasa J, Villaret M. The RANTANPLAN planner: system description. The Knowledge Engineering Review (KER). 2016;31(5):452–464.

[22] Audemard G, Simon L. Predicting Learnt Clauses Quality in Modern SAT Solvers. In: IJCAI; 2009. p. 399–404.

[23] Brummayer R, Biere A. Boolector: An Efficient SMT Solver for Bit-Vectors and Arrays. In: TACAS; 2009. p. 174–177.

[24] Nightingale P, Akgün Ö, Gent IP, Jefferson C, Miguel I. Automatically improving constraint models in Savile Row through associative-commutative common subexpression elimination. In: CP; 2014. p. 590–605.

[25] Nightingale P, Spracklen P, Miguel I. Automatically improving SAT encoding of constraint problems through common subexpression elimination in Savile Row. In: CP; 2015. p. 330–340.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210118

A Cognitively-Inspired Model for Making Sense of Hasse Diagrams

Dimitra BOUROU [a,b,1] Marco SCHORLEMMER [a,b] and Enric PLAZA [a]
[a] *Artificial Intelligence Research Institute, IIIA-CSIC, Bellaterra, Catalonia, Spain*
[b] *Dept. Ciències de la Computació, Universitat Autònoma de Barcelona, Bellaterra, Catalonia, Spain*

Abstract.
In this paper, we present a model of the sense-making process for diagrams, and describe it for the case of Hasse diagrams. Sense-making is modeled as the construction of networks of conceptual blends among image schemas and the diagram's geometric configuration. As a case study, we specify four image schemas and the geometric configuration of a Hasse diagram, with typed FOL theories. In addition, for the diagram geometry, we utilise Qualitative Spatial Reasoning formalisms. Using an algebraic specification language, we can compute conceptual blends as category-theoretic colimits. Our model approaches sense-making as a process where the image schemas and the diagram geometry both structure each other through a complex network of conceptual blends. This yields a final blend in which the sort of inferences we confer to diagrammatic representations emerge. We argue that this approach to sense-making in diagrams is more cognitively apt than the mainstream view of a diagram being a syntactic representation of some underlying logical semantics. Moreover, our model could be applied to various types of stimuli and is thus valuable for the general field of AI.

Keywords. diagrammatic reasoning, conceptual blending, conceptual meaning, image schema, first-order logic, formal specification, sense-making.

1. Introduction

Sense-making refers to the process by which we structure our percepts into constructs that are more meaningful for us. In this work, we model the sense-making of diagrams as conceptual blends of their geometric configurations, with image schemas. The latter reflect early embodied sensorimotor experiences [1,2]. To the best of our knowledge, modeling the sense-making of diagrams in this manner is a novel contribution, which could be of value for fields pertaining to human-human or human-machine communication by means of graphical aids.

To illustrate our approach, take for instance the Hasse diagram of Fig. 1 (left). Its geometric configuration comprises points and lines, with each line intersecting with one pair of points. We propose that some of the possible ways one could make sense of this diagram are that points h, e, b and a form a path with direction from h to a, or that h with

[1] Corresponding Author: Dimitra Bourou, Artificial Intelligence Research Institute (IIIA-CSIC), Campus UAB, Carrer de Can Planes, Zona 2, 08193 Bellaterra, Barcelona, Spain; e-mail: dbourou@iiia.csic.es

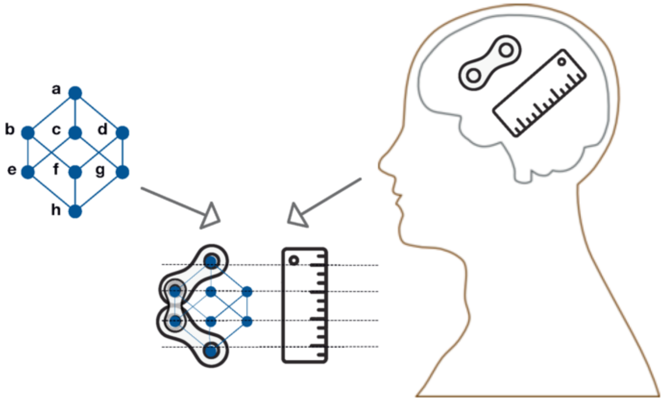


Figure 1. Conceptual blend of a Hasse diagram. The LINK or SCALE image schemas (right) are blended with the geometric configuration (left), yielding the Hasse diagram as we make sense of it (bottom).

e, e with b, and b with a, form pairs of entities symmetrically linked by lines. The diagram may also be understood as a scale of four grades, each grade consisting of the points on the same horizontal axis, e.g., grade 2 would comprise points e, f and g. Finally, some or all the above conceptualisations could be applied jointly. These conceptualisations allow the emergence of some inferences on the diagram, such as h and a being associated, as parts of the same path, or d being transitively on a higher grade than h.

This variety of understandings of the diagram leads to different conclusions, depending on whether the ‘path’, ‘link’, or ‘scale’ conceptualisation, or all of them, are at play; each imbues a different sense on the diagram, i.e., direction, symmetric association, and quantity, respectively. This shows that diagrams, taken as geometric configurations, do not bring up a unique way of making sense of them. We believe these alternative senses can be modeled as a result of combining different image schemas with each other and with the geometric configuration. The aforementioned concepts correspond directly with their homonymous image schemas and can thus be formalised as such. Appropriate correspondences between the two schemas, and of the schemas with the geometric configuration, are responsible for constraining the exact form of the image schemas, or their blends, that will contribute to structuring the geometry. For example, the instance of SCALE corresponding to the Hasse diagram of Fig. 1 would have four grades. Subsequently, given these correspondences, the final blend between image schemas and geometry can be constructed, and inferences can arise thereby.

Our proposal is to model the aforementioned sense-making process as follows: A configuration consists only of geometric entities. The ability to do inference with the diagram is not a result merely of its geometry; the sense-making, which makes inference possible, arises when image schemas, with their internal structure, are blended with each other, and with the geometry, structuring it into a meaningful diagram (Fig. 1). Here, we describe our model of sense-making for the case of Hasse diagrams. We believe that viewing sense-making as emerging from an entire network of interrelated concepts paves the way towards an efficient and cognitively plausible model of conceptual meaning. Such a model would comprise only a few image schemas and blending principles, but would be able to give rise to a host of different instantiations of image schemas and various interpretations for a given stimulus.

2. Background

The literature of diagrammatic reasoning has been valuable for formally studying the informational content and the efficacy of diagrams for inference. In order to reach such conclusions, a one-to-one correspondence between the geometry (syntax) and the semantics of the diagram is typically assumed. However, as explained above, a certain geometric configuration does not always evoke a unique understanding. Furthermore, the interpretation of diagrams entails a constructive and imaginative process [3]. This is in line with the process of sense-making, which we model computationally in this work.

Sense-making is defined in the literature of enactive cognition as the process of an autonomous agent bringing its own original meaning upon its environment [4]. Image schemas are fundamental for such a process, because they have the capacity to organise and structure our experience [2, p. 372]. Image schemas are mental structures, formed early in life, that constitute structural contours of repeated sensorimotor contingencies such as SUPPORT, VERTICALITY, and BALANCE. For example, the meaning of balance emerges through the repeated embodied experience of different kinds of balance, which leads to the formation of a mental structure reflecting what is invariant among them [1, p. 74-75]. Image schemas are gestalts; they consist of components in a specific relational structure, which can be projected onto another domain and guide inference in it. This is related to the phenomenon of mental visualization, i.e., seeing in our "mind's eye" a generic chair when reading the word 'chair' [5,6]. Mental visualization is necessary for inference and prediction, and image schemas have been proposed to enable such visualization [7, pp. 513, 519-520].

One way to approach the aforementioned projection of image schema structure onto other domains is through conceptual blending. Conceptual blending is a process by which several mental spaces —coherent and integrated chunks of information that underlie cognition and which comprise entities, and relations or properties that characterise them— are put into correspondence with each other via cross-space relations, so as to be integrated into a blend with novel structure [8]. The original mental spaces can be referred to as input spaces, and the resulting blend as blended space.

In Fig. 1, the input spaces involved are the Hasse geometric configuration and the LINK and SCALE schemas. The blend integrates both image schemas with the geometrric configuration, capturing one way we can make sense of the diagram, i.e., as comprising a set of symmetrically associated entities (owing to LINK), and, at the same time, a single graded directional structure (owing to SCALE).

3. Approach

In this paper, we model the sense-making of a sensory stimulus, i.e., a diagram, by means of cross-space correspondences between an input space reflecting this stimulus, and input spaces reflecting instances of image schemas. These correspondences guide the construction of blended spaces, integrating image schemas with some substructure of the stimulus. The emergent, integrated structure of this blended space enables novel inferences that were not possible in each input space alone. The entirety of the complex network of correspondences between spaces, and the blends emerging, reflect the sense-making of the given stimulus (here, geometric configuration).

To model this network of blends, we provide: (a) A formal specification of the geometry of the diagram. Qualitative Spatial Reasoning (QSR) formalisms model several aspects of spatial configurations at a level compatible with human perception. Here, we use QSR and typed first-order logic (FOL) theories to capture the geometry of the diagram. (b) Formal specifications of the structure of the relevant image schemas, also by means of typed FOL theories. (c) A formalisation of suitable correspondences between image-schema structures and the geometric configuration. Given such correspondences, each blend can be modeled as a particular kind of category-theoretical colimit [9].

3.1. Diagrammatic notation and geometric configuration

A Hasse diagram represents a partially ordered set (poset). It consists of edges and vertices, drawn as points and lines. Each point represents one element of the poset. Assuming elements x, y and z of the poset, ordered by the '$<$' relation, then the syntactic rules are that if $x < y$, then x is shown in a lower position than y in the diagram, and that x and y are connected by a line in the diagram iff: $x < y$ or $y < x$, and there is no element z such that $x < z$ and $z < y$.

In order to describe this geometric configuration, we draw from formal systems of the QSR literature. Existing formalisms enable us to characterise spatial entities as points, lines, and regions, to describe their topological relations [10], and their qualitative position with respect to each other [11]. Concretely, the Hasse configuration of Fig. 1 has eight points (a to h) and twelve lines (ba, eb, etc.). Each line intersects with a pair of points. Below is a fragment of the specification of this configuration, which states some of the topological and orientation relations of its geometric entities.[2]

$$\begin{array}{lcr}
\mathit{intersects}(ba,a) & \mathit{intersects}(ba,b) & \mathit{right\_front}(a,b) \\
\mathit{intersects}(eb,b) & \mathit{intersects}(eb,e) & \mathit{front}(b,e) \\
\mathit{intersects}(he,e) & \mathit{intersects}(he,h) & \mathit{left\_front}(e,h)
\end{array}$$

3.2. Image schema formalisation

To capture the structure of image schemas, we formalise their components and logical properties as typed FOL theories. In this effort, we were guided by conceptual descriptions of image schemas (mainly [1,2]), or experimental work, when available. For lack of space, we present here only the LINK and the VERTICALITY schemas in formal detail, and the remaining schemas in brief.[3]

[2] Predicates such as *intersects* state the topological relations as defined in [10], while predicates such as *right_front* state the orientation of entities as defined in [11]. Constants a to h are of type *Point*; constants *ba*, *eb*, etc. are of type *Line*. We note that the formalism proposed by Hernandez [11] offers various possible reference frames for orientation. These include frames depending on object functionalities, observers' viewpoint, or global, absolute frames like *north*, *west* etc. The latter is the approach we have followed, approaching all diagrams as they are typically presented on paper, with the upper part being *front*. Moreover, all QSR formalisms allow spatial reasoning. However, our goal here is to explore the reasoning afforded by blending the geometry with image schemas, and thus use the QSR system only for representation purposes.

[3] The complete formalisation of the blends modeling the sense-making of the Hasse diagram, and more kinds of diagrams, can be downloaded from `https://drive.google.com/drive/folders/1jcQdJTOqbnAua3uXIgTEW8zV3kF_2R14?usp=sharing`.

LINK. The LINK schema pertains to the notion of association, either physical or abstract. The prototypical LINK schema associates two distinct, usually contiguous, entities with each other through a link. The relevant axioms are presented and explained below.[4]

$$\forall s \in \mathit{LinkSchema} : \mathit{linked}(\mathit{anEnt}(s), \mathit{anotherEnt}(s))$$

$$\forall x,\ y \in \mathit{Entity}; s \in \mathit{LinkSchema}\ :\ \mathit{linked}(x,y) \Leftrightarrow$$
$$(\mathit{anEnt}(s) = x \ \wedge\ \mathit{anotherEnt}(s) = y)\ \vee\ (\mathit{anEnt}(s) = y \ \wedge\ \mathit{anotherEnt}(s) = x)$$

$$\forall l \in \mathit{Link}\ \exists ! s \in \mathit{LinkSchema}\ :\ l = \mathit{link}(s)$$

$$\forall x,\ y \in \mathit{Entity} :\ \mathit{linked}(x,y) \Leftrightarrow \mathit{linked}(y,x)$$

$$\forall x \in \mathit{Entity} : \neg \mathit{linked}(x,x)$$

PATH. The PATH schema is related to directionality and motion from a source towards a goal. It comprises a non-branching series of adjacent locations which connect the source with the goal. By the structure of the schema, it is obvious that, if someone is on a certain location of the path, then they have already traversed all prior locations. Therefore, the PATH schema is axiomatised as a total order; a collection of serially neighboring locations with the source and goal as endpoints.

VERTICALITY. The VERTICALITY schema obtains its structure from our experience of standing upright with our bodies resisting to gravity, or from perceiving upright objects like trees. The VERTICALITY schema reflects the axis of an upright object, so it must have a base as the bottom of the object [12]. Moreover, the axis may be merely mentally visualized by the observer; for example, when observing the sun, the horizon is the base, and a visualized vertical axis runs upward from the horizon, reaching the sun. Consequently, VERTICALITY is modeled simply as a unique vertical axis with its base. The base is a mark on the axis, such that no other mark on the axis can be placed above it.

$$\forall s \in \mathit{VerticalitySchema} :\ \mathit{inAxis}(\mathit{base}(s), \mathit{axis}(s))$$

$$\forall m \in \mathit{Mark} :\ \neg \mathit{above}(m,m)$$

$$\forall s \in \mathit{VerticalitySchema}; m \in \mathit{Mark} : \mathit{inAxis}(m, \mathit{axis}(s))\ \wedge\ (m \neq \mathit{base}(s)) \Rightarrow \mathit{above}(m, \mathit{base}(s))$$

SCALE. The SCALE schema relates to a gradient of quantity. We believe that it comprises an ordered set of several grades, but, unlike VERTICALITY, it should not imply a particular geometric orientation. The SCALE schema has a directionality, and a cumulative property; if one has 15 euros, they also have 10. This structure is modeled as a total order on grades.

3.3. A formal model of sense-making

Given the aforementioned formalisation of the input spaces involved, the sense-making of the Hasse configuration is modeled as the complex conceptual blending of several simpler blends.

[4] Elements of type *LinkSchema* are constituted of two components of type *Entity* and one component of type *Link*, which are obtained with functions *anEnt*, *anotherEnt*, and *link*, respectively (not shown). The axioms above state that that the two entities of a *LinkSchema* are always linked; that linked entities are always part of some *LinkSchema*; that a link is always part of a unique *LinkSchema*; and that the *linked* predicate is irreflexive and symmetric.

One part of the network of blends at hand includes the input spaces of the PATH and the LINK schemas (Fig. 2). To blend these two schemas with each other, and with the geometric configuration, each pair of linked entities, and each pair of contiguous locations in a path, are put into correspondence with two points of the geometric configuration which intersect with the same line. By extension, a sequence of points connected by lines in the Hasse configuration (e.g., points h, e, b, and a in Fig. 1; left), can be put into correspondence with a particular instance of PATH by relating connected points (such as h and e, b and e, and b and a) with contiguous locations of PATH, and end points (such as h and a) with the source and the goal of PATH respectively. More precisely, these cross-space correspondences between the input spaces of the Hasse geometric configuration and the LINK or the PATH schema can be expressed as pairs of a binary relation R between entities of the two spaces. For example, the points h, e, b, and a are related through R with the *source*, l_2, l_3, and *goal* locations of an instance of PATH as follows:

$$R(source,h) \qquad R(l_2,e) \qquad R(l_3,b) \qquad R(goal,a)$$

All the aforementioned correspondences allow the construction of a network of blends (Fig. 2) and ultimately result in the integration of LINK, PATH, and the geometric configuration, into a final blend comprising an integrated geometric and image-schematic structure with both directed paths of consecutive points, as well as pairs of linked points. Notice that the structure of the geometric configuration contributes to shaping the precise form of the image schema instances, and how they are blended with each other. At the same time, as all input spaces (image schemas and geometry) are involved in an intricate network of correspondences and blends (Fig. 2), they all structure each other, resulting in a final blend that comprises an integration of structure from all input spaces.

In a similar way, the VERTICALITY and SCALE schemas also structure the Hasse diagram. The marks of an instance of VERTICALITY, and the grades of an instance of SCALE schema, are put into correspondence with points of the Hasse diagram that are in the same horizontal axis. Point h, which is geometrically lowest, corresponds to the base of VERTICALITY. Marks, or grades whereby one is ordered immediately above the other, correspond to pairs of points whereby one is oriented immediately 'front' (or 'left_front', or 'right_front') of the other. This blend comprises a single 'vertical scale', with four integrated marks-grades (henceforth called 'levels'), the lowest of which is the base. Points belong to these levels and can thus be oriented with respect to the down-up axis. The configuration is thus now imbued with both the sense of verticality, as well as the sense of quantity, from the VERTICALITY and SCALE schemas respectively.

In summary, the Hasse diagram, as we make sense of it, emerges from a complex conceptual network with the four aforementioned image schemas, and the Hasse geometry, as input spaces. The correspondences between various instances of these schemas and substructures of the Hasse configuration, yield the Hasse diagram as comprising several paths of linked points, arranged at several levels of generality along an upward vertical axis. Some of these elements of different paths are on the same level of generality. Moreover, there is a unique source ordered before all other elements of the diagram, and a unique goal ordered after all other elements. Mathematically, these blends are computed as category-theoretic colimits of typed FOL theories [9].

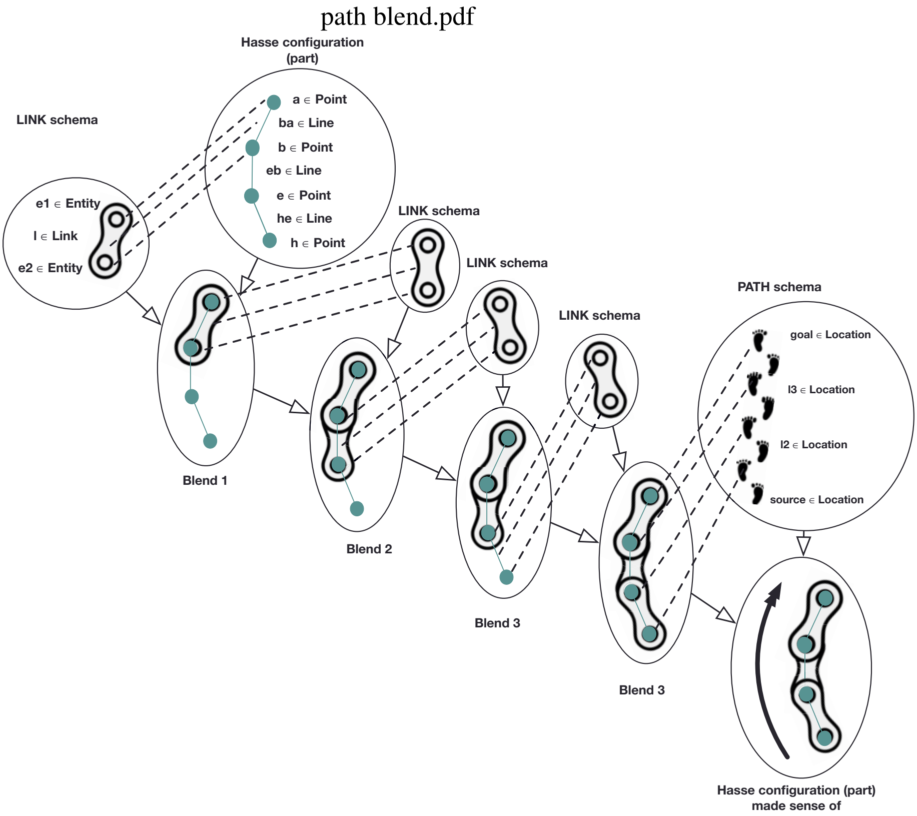


Figure 2. The network of blends that integrates the LINK and PATH schemas with part of the Hasse configuration. Correspondences between elements are shown with dashed lines.

4. Related work

In diagrammatic reasoning it is posited that the efficacy of diagrams lies in their sharing structural properties with their referents. These properties allow the observers to make inferences with these diagrams [13]. Therefore, the more the properties of the geometry of a diagram match the properties of its semantics, the more efficacious this diagram would be to represent this semantics, and to lead to valid inferences. Here, we expanded in this direction by modeling the origin of these properties as the blending of image schemas with the geometry of a diagram.

A few research groups have worked on formalising image schemas and the relations among them. Rodriguez and Egenhofer [14] provide a relational algebra inspired by the CONTAINER and SURFACE schemas, used to model, and reason about, spatial relations of objects in an indoor scene. Image schemas have also been used to model planning and actions of agents [15]. In the latter work, some image schemas were recursively defined in terms of other schemas. In both these works, the formalisations are merely inspired by image schemas, rather than faithful representations of their descriptions in the literature. Kuhn [16] formalised image schemas, and their combinations, as ontology relations using functional programming, in a relatively abstract manner. In a recent, comprehensive work, Hedblom [17] modeled image schemas as families of interrelated logical theories, with each schema comprising a combination of primitive components. QSR formalisms that capture the spatiotemporal content of schemas were used. In the present approach, we chose not to use such formalisms to capture the internal structure of image schemas.

Image schemas and blending have been used jointly mostly to model creative processes. Schorlemmer et al. [18] modeled the creative problem-solving process of tackling a riddle by way of a category-theoretic characterisation of blending, based on typed FOL theories of image schemas. In this work, image schemas were used to establish shared structure between different input spaces. In contrast, in a conceptual work by Falomir et al. [19], image schemas are used as input spaces, together with a QSR description of an icon, in order to blend them to interpret the latter. Importantly, here the stimulus mediates and structures the blending of image schemas with each other, as in our work. Finally, Embodied Construction Grammar allows the formalisation [20] and implementation [21] of language understanding by mapping components of specific schemas (image schemas, and others) to phonemes. The last two works are analogous to our own, modeling the sense-making of diagrams; except the stimulus made sense of is an icon and a spoken sentence respectively, instead of a diagram.

5. Discussion

In this paper we have presented a formal framework of the sense-making of diagrams as observers mentally structuring the geometry of diagrams by unconsciously projecting preexisting mental structures — i.e., image schemas — onto it, giving rise to inferences. We described examples of such inferences and pointed out that they are not fully determined by the geometry of the diagram, but also by image schemas in the observer's mind, and the way they are put into correspondence with this geometry.

Applying our framework to a Hasse diagram, we observe that the following facts can all be inferred from the geometric configuration : (a) point a is above point h (b) points h, e, b and a form a path and (c) points b, c and d are on the same level. To make inference (a), for instance, an observer may mentally visualize a physical path of linked locations, starting at location h, extending towards higher locations e and b, up to a, which lies above h and the rest of the locations traversed in the path. This mental visualization facilitates making the inference that $h < a$ directly from the Hasse diagram. Mental visualization is necessary for inference, and image schemas are the mental structures that enable it [7, pp. 513, 519]. Subjects are indeed able to make a correct one-step transitive inference on a given vertically-oriented diagram, without physically manipulating it [22], so mental visualization could be involved. More generally, the diagrammatic inferences that are captured in our blend network are: the transitive ordering of points in terms of their level, the inference that the source and the goal point of the PATH schema are ordered before and after all other points, respectively (corresponding semantically to the minimal and maximal element), and the existence of distinct instances of PATH schema (including all maximal chains).

Hasse diagrams indeed prioritise visualizing the structure of the order they represent, through a vertical organization, and explicit visualization of levels[5] [23]. Following the assumption that the efficacy of diagrams results from shared properties between

[5] A Hasse diagram explicitly shows levels when elements with the same rank in the poset, i.e., same number of steps away from the minimum element, are placed in the same horizontal axis.

their geometry and semantics [13], a Hasse diagram would be efficacious to represent a poset, because both its geometric configuration of shapes along a vertical axis, as well as its partial order semantics, share properties of transitivity and asymmetry. In our view, these properties may become cognitively salient by integrating —i.e., building cross-space correspondences between— the VERTICALITY and SCALE schemas with both the Hasse configuration and the poset semantics. In fact, it has been proposed that our understanding of abstract set theoretical notions also rests on image schemas [24]. The aforementioned inferences (transitivity, existence of maximal elements, minimal elements, and maximal chains) can be made through a process of integrating image schemas with the geometry and with the semantics of a Hasse diagram. This view is further supported by experiments showing that Hasse diagrams that are not upright, or do not show levels, take longer to be interpreted by subjects [25]. The precise role of the diagram semantics in the blend networks representing sense-making will be explored in future work.

The novelty of our framework lies in the fact that it is not merely conceptual but also written in a formal, computer-processable language. We contribute to the literature with a reusable set of formalized image schemas. Importantly, our framework is general enough to apply to any type of stimuli that is expressible in typed FOL. Furthermore, the entire framework could eventually be generalized in a representation-independent manner as described in [9]. Finally, by formalising the optimality principles —proposed to guide the construction of networks of blends that are desirable, i.e., lead to emergent structure that is useful for inference— we could obtain possible cross-space correspondences automatically, as in [26]. Such a framework would comprise an efficient and cognitively plausible model of sense-making, which we consider a valuable contribution to AI.

As for diagrams, the issue of generating and evaluating alternative blends for a given configuration, including those that model erroneous interpretations, i.e., inconsistent ones with the intended semantics, will be explored in the future. Such work would allow us to pin down what makes one diagram likely to be more accurately interpreted than another. This information could be of value for human-computer interaction because it could provide guidelines for the design of efficacious diagrammatic and graphical visualizations. For example, if a designer wants to visually represent some ordinal values, a tool based on our framework might recommend the use of a vertical geometric configuration and not a horizontal one. This is because a VERTICAL-SCALE is likely to map to such a configuration and lead to a blend with the intended semantics. Various such recommendations can be made precise thanks to our model and could contribute to new tools directed at designers.

6. Acknowledgments

The present research was supported by CORPORIS (PID2019-109677RB-I00) funded by Spain's *Agencia Estatal de Investigación*; by DIAGRAFIS (202050E243) funded by CSIC; and by the *Ajuts a grups de recerca consolidats* (2017 SGR 172) funded by *AGAUR / Generalitat de Catalunya*.

References

[1] Johnson M. The Body in the Mind: The Bodily Basis of Meaning, Imagination, and Reason. University of Chicago Press; 1987.

[2] Lakoff G. Women, Fire, and Dangerous Things. University of Chicago Press; 1987.

[3] May M. Diagrammatic Reasoning and Levels of Schematization. In: Iconicity. A Fundamental Problem in Semiotics. NSU Press, Copenhagen; 1999. p. 175–194.

[4] Varela FJ. Organism: A Meshwork of Selfless Selves. In: Organism and the Origins of Self. Springer; 1991. p. 79–107.

[5] Jackendoff R. Semantics and Cognition. vol. 8. MIT Press; 1983.

[6] Jackendoff R. Consciousness and the Computational mind. The MIT Press; 1987.

[7] Mandler JM, Cánovas CP. On defining image schemas. Lang Cogn. 2014;6(4):510–532.

[8] Fauconnier G, Turner M. The Way We Think. Basic Books; 2002.

[9] Schorlemmer M, Plaza E. A uniform model of computational conceptual blending. Cogn Syst Res. 2021;65:118–137.

[10] Egenhofer MJ, Herring JR. Categorizing binary topological relations between regions, lines, and points in geographic databases. Department of Surveying Engineering, University of Maine; 1991.

[11] Hernández D. Relative Representation of Spatial Knowledge: The 2-D Case. In: Cognitive and Linguistic Aspects of Geographic Space. Springer; 1991. p. 373–385.

[12] Serra Borneto C. Liegen and Stehen in German: A study in horizontality and verticality. In: Cognitive Linguistics in the Redwoods. Mouton de Gruyter; 1996. p. 459–506.

[13] Shimojima A. On the efficacy of representation [PhD dissertation]. Indiana University; 1996.

[14] Rodríguez MA, Egenhofer MJ. A comparison of inferences about containers and surfaces in small-scale and large-scale spaces. J Vis Lang Comput. 2000;11(6):639–662.

[15] St Amant R, Morrison CT, Chang YH, Mu W, Cohen PR, Beal C. An image schema language. North Carolina State University at Raleigh, Dept. of Computer Science; 2006.

[16] Kuhn W. An image-schematic account of spatial categories. In: Proc. International Conference on Spatial Information Theory. Springer; 2007. p. 152–168.

[17] Hedblom MM. Image schemas and concept invention: Cognitive, logical, and linguistic investigations. Springer; 2020.

[18] Schorlemmer M, Confalonieri R, Plaza E. The Yoneda Path to the Buddhist Monk Blend. In: Proc. of the Joint Ontology Workshops; 2016. .

[19] Falomir Z, Plaza E. Towards a model of creative understanding: Deconstructing and recreating conceptual blends using image schemas and Qualitative Spatial Descriptors. Ann Math Artif Intell. 2019;88:457–477.

[20] Bergen B, Chang N. Embodied construction grammar in simulation-based language understanding. In: Construction grammars: Cognitive grounding and theoretical extensions. vol. 3. John Benjamins; 2005. p. 147–190.

[21] Bryant JE. Best-Fit Constructional Analysis [PhD dissertation]. EECS Department, University of California, Berkeley. Magdeburg, Germany; 2008. http://www2.eecs.berkeley.edu/Pubs/TechRpts/2008/EECS-2008-100.html.

[22] Shimojima A, Katagiri Y. An eye-tracking study of exploitations of spatial constraints in diagrammatic reasoning. Cogn Sci. 2013;37(2):211–254.

[23] Demey L, Smessaert H. The relationship between Aristotelian and Hasse diagrams. In: Proc. of the International Conference on Theory and Application of Diagrams. Springer; 2014. p. 213–227.

[24] Lakoff G, Núñez RE. Where mathematics comes from: How the embodied mind brings mathematics into being. AMC. 2000;10(12):720–733.

[25] Körner C, Albert D. Comprehension efficiency of graphically presented ordered sets. In: Current psychological research in Austria. Proc. of the 4th Scientific Conference of the Austrian Psychological Society. Graz: Akademische Druck - u. Verla; 2001. p. 179–182.

[26] Pereira FC, Cardoso A. Optimality principles for conceptual blending: A first computational approach. AISB J. 2003;1(4):351–369.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210119

Some Advances on the Solution of the Generalized Law of Importation for Fuzzy Implication Functions

Isabel AGUILÓ [a,b,1], Sebastia MASSANET [a,b] and Juan Vicente RIERA [a,b]
[a] *SCOPIA research group, Dept. of Mathematics and Computer Science, University of the Balearic Islands, 07122 Palma, Spain*
[b] *Health Research Institute of the Balearic Islands (IdISBa), 07010 Palma, Spain*

Abstract. The law of importation has attracted the interest of many researchers devoted to fuzzy implication functions in the last decades. This property has several important applications, especially in approximate reasoning and image processing. Several generalizations of this property have been proposed. Specifically, one generalization related to the law of migrativity was recently introduced by Baczyński et al. in which two fuzzy implication functions are involved. In this paper, some advances on the solution of this functional equation for the particular case where the involved fuzzy conjunction is a t-norm are presented. Indeed, a complete characterization of all those pairs of fuzzy implication functions with a strict natural fuzzy negation satisfying the generalized law of importation is achieved.

Keywords. Fuzzy implication function, T-norm, Functional equation, Generalized law of importation, Natural negation

1. Introduction

One of the main lines of research in fuzzy logic is the theoretical study of fuzzy implication functions. These operators, which are the generalization of the classical implication to the fuzzy framework, have important applications in approximate reasoning or image processing [2]. For a concrete application, fuzzy implication functions fulfilling some additional algebraical properties are needed and therefore, the characterization of such operators (obtained by solving functional equations) provides a bunch of operators useful for those applications.

One of such additional properties is the *law of importation* with respect to a fuzzy conjunction C. This property is given by

$$I(C(x,y),z) = I(x,I(y,z)) \quad \text{for all} \quad x,y,z \in [0,1], \ (\textbf{LI}) \tag{1}$$

where I is a fuzzy implication function and C a fuzzy conjunction (for more details see [1,6]). The significance of the law of importation is related to the simplification of the

[1] Corresponding Author: Isabel Aguiló, Dept. of Mathematics and Computer Science, University of the Balearic Islands, Ctra. de Valldemossa, Km.7.5, 07122 Palma, Spain; E-mail: isabel.aguilo@uib.es.

process of applying the Compositional Rule of Inference (CRI) of Zadeh reducing its complexity through the so-called Hierarchical CRI [4]. In [3], Baczyński et al. discuss a generalization of the laws of α-migrativity, which leads to a generalized version of the law of importation. In this way, the triple (C,I,J) is said to satisfy the *generalized law of importation* if it satisfies

$$I(C(x,y),z) = I(x,J(y,z)) \quad \text{for all} \quad x,y,z \in [0,1], \qquad (\textbf{GLI})$$

where I,J are fuzzy implication functions and C a fuzzy conjunction. It is straightforward to check that if $I = J$, then the standard law of importation is retrieved. In [3], an initial study of the functional equation is presented studying in depth its fulfillment when the pairs (I,C) or (J,C) satisfy the law of importation. However, these conditions are not necessary and the characterization of the solutions remains an open problem.

In this paper, we present some advances on the characterization of the solutions for the particular case in which a t-norm is considered as fuzzy conjunction and the two involved fuzzy implication functions have strict natural negations. This is an ongoing study with the final goal of characterizing all pairs of fuzzy implication functions with continuous natural negations which satisfy **(GLI)** with a t-norm.

2. Preliminaries

For the sake of completeness, we recall the definitions of t-norms, fuzzy implication functions and fuzzy negations.

Definition 1 ([1]). *A function $T : [0,1]^2 \to [0,1]$ is called a* triangular norm *(shortly t-norm) if T is associative, commutative, increasing in each variable and has neutral element* 1.

Definition 2 ([1]). *A binary operator $I : [0,1] \times [0,1] \to [0,1]$ is said to be a* fuzzy implication function *if it satisfies:*

(I1) *$I(x,z) \geq I(y,z)$ when $x \leq y$, for all $z \in [0,1]$.*

(I2) *$I(x,y) \leq I(x,z)$ when $y \leq z$, for all $x \in [0,1]$.*

(I3) *$I(0,0) = I(1,1) = 1$ and $I(1,0) = 0$.*

Note that, from the definition, it follows that $I(0,x) = 1$ and $I(x,1) = 1$ for all $x \in [0,1]$ whereas the values $I(x,0)$ and $I(1,x)$ are not derived.

Definition 3 ([1]). *A decreasing function $N : [0,1] \to [0,1]$ is called a fuzzy negation if $N(0) = 1$ and $N(1) = 0$. Moreover, a fuzzy negation is strict if it is strictly decreasing and continuous.*

The natural negation of a fuzzy implication function will be also useful in our study.

Definition 4 ([1]). *Let I be a fuzzy implication function. The function N_I defined by $N_I(x) = I(x,0)$ for all $x \in [0,1]$, is called the* natural negation *of I.*

3. The generalised law of importation: a characterization

In this section we will prove the characterization of all pairs of fuzzy implication functions with strict natural negations satisfying the generalized law of importation.

Proposition 1. *Let $I, J : [0,1]^2 \to [0,1]$ be two binary functions with N_I and N_J strict negations and T be a t-norm. If the triple (I,J,T) satisfies* (**GLI**) *then for all $x,y \in [0,1]$,*

$$I(x,y) = N_I(T(x, N_J^{-1}(y))), \tag{2}$$

$$J(x,y) = N_J(T(x, N_J^{-1}(y))). \tag{3}$$

Proof. First of all, note that N_J is a strict fuzzy negation and, consequently, $Ran(N_J) = [0,1]$. Then, in order to prove Expression (2), it is enough to check that

$$I(x, N_J(y)) = N_I(T(x, N_J^{-1}(N_J(y)))).$$

Thus,

$$I(x,N_J(y)) = I(x, J(y,0)) = I(T(x,y),0) = N_I(T(x,y)) = N_I(T(x, N_J^{-1}(N_J(y)))).$$

On the other hand, to prove Expression (3), note that

$$I(x,y) = I(T(1,x),y) = I(1, J(x,y)) = N_I(T(1, N_J^{-1}(J(x,y))) = N_I \circ N_J^{-1}(J(x,y))$$

and then $J(x,y) = N_J \circ N_I^{-1}(I(x,y)) = N_J(T(x, N_J^{-1}(y)))$ for all $x,y \in [0,1]$. ■

Next result shows that if we consider two binary functions $I, J : [0,1]^2 \to [0,1]$ whose expressions are given by Formulas (2) and (3) respectively, then the triple (I,J,T) fulfills the generalized law of importation.

Proposition 2. *Let T be a t-norm and let $I, J : [0,1]^2 \to [0,1]$ be two binary functions whose expressions are given by Formulas (2) and (3), respectively, where N_I and N_J are strict negations. Then, the triple (I,J,T) fulfills* (**GLI**).

Proof. Suppose that binary operations I and J are given by Formulas (2) and (3), respectively. Then, for all $x,y,z \in [0,1]$, we have that

$$\begin{aligned} I(T(x,y),z) &= N_I(T(T(x,y), N_J^{-1}(z))) = N_I(T(x, T(y, N_J^{-1}(z)))) \\ &= N_I(T(x, N_J^{-1}(N_J(T(y, N_J^{-1}(z)))))) = I(x, J(y,z)). \end{aligned}$$

■

Taking into account the two previous results, the next theorem follows.

Theorem 1. *Let T be a t-norm and $I, J : [0,1]^2 \to [0,1]$ be two binary functions with N_I and N_J strict negations. Then, the triple (I,J,T) satisfies* (**GLI**) *if and only if,*

$$I(x,y) = N_I(T(x, N_J^{-1}(y))),$$

$$J(x,y) = N_J(T(x, N_J^{-1}(y))).$$

Proof. It follows immediately from Propositions 1 and 2. ■

The following example illustrates the previous Theorem 1.

Example 1. *Let us consider the strict negations $N_J(x) = 1 - x$ and $N_I(x) = 1 - x^2$ for all $x \in [0,1]$ and the product t-norm $T_{\mathbf{P}}(x,y) = xy$ for all $x,y \in [0,1]$. In this case, applying Formulas (2) and (3), we obtain that the only fuzzy implication functions with such natural negations satisfying* **(GLI)** *with $T_{\mathbf{P}}$ are given by*

$$\begin{aligned} I(x,y) &= 1 - x^2 + 2yx^2 - x^2y^2, \\ J(x,y) &= 1 - x + xy \text{ (the Reichenbach implication).} \end{aligned}$$

4. Conclusions and Future Work

In this paper, we have characterized all the couples of fuzzy implication functions (I,J) fulfilling $(\mathbf{GLI})$ with a t-norm T when their natural negations are strict. We have shown that the expressions of these fuzzy implication functions can be obtained from their natural negations N_I and N_J and from the considered t-norm T. As a future work, we wish to continue the study in the case that the natural negations are continuous (but not necessarily strict) and, in a more general framework, we want to consider a conjunctive uninorm instead of a t-norm.

Acknowledgment

This work has been partially supported by the Spanish Grant FEDER/Ministerio de Economía, Industria y Competitividad-AEI/TIN2016-75404-P.

References

[1] M. Baczyński, B. Jayaram, Fuzzy Implications. Studies in Fuzziness and Soft Computing, vol. 231. Springer, Berlin Heidelberg, 2008.

[2] M. Baczyński, B. Jayaram, S. Massanet, and J. Torrens, "Fuzzy implications: Past, present, and future", in Springer Handbook of Computational Intelligence, J. Kacprzyk and W. Pedrycz, Eds. Springer Berlin Heidelberg, 2015, pp. 183–202.

[3] M. Baczyński, B. Jayaram and R. Mesiar, Fuzzy Implications: alpha migrativity and generalised laws of importation, Information Science. 2020; 531: 87-96.

[4] B. Jayaram, On the law of importation $(x \wedge y) \rightarrow z \equiv (x \rightarrow (y \rightarrow z))$ in fuzzy logic, IEEE Transactions on Fuzzy Systems. 2008; 16: 130–144.

[5] M. Mas, M. Monserrat, J. Torrens, E. Trillas, A survey on fuzzy implication functions,IEEE Transactions on Fuzzy Systems. 2007; 15(6): 1107-1121.

[6] S. Massanet, J. Torrens. Characterization of Fuzzy Implications Functions With a Continuous Natural Negation Satisfying the Law of Importation With a Fixed t-Norm. IEEE Transactions on Fuzzy Systems. 2017;25(1):100-113.

[7] S.Massanet, J.Torrens. The law of Importation versus the exchange principle on fuzzy implications. Fuzzy Sets Systems. 2011; 168 (1): 47-69.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210120

Enabling Game-Theoretical Analysis of Social Rules

Nieves MONTES [a,1], Nardine OSMAN [a] and Carles SIERRA [a]
[a] *Artificial Intelligence Research Institute (IIIA-CSIC)*
Campus de la UAB, 08193 Bellaterra (Barcelona)

Abstract. In the field of normative multiagent systems, the relationship between a game structure and its underpinning agent interaction rules is hardly ever addressed in a systematic manner. In this work, we introduce the Action Situation Language (ASL), inspired by Elinor Ostrom's Institutional Analysis and Development framework, to bridge the gap between games and rules. The ASL provides a syntax for the description of agent interactions, and is complemented by an engine that automatically provides semantics for them as extensive-form games. The resulting games can then be analysed using standard game-theoretical solution concepts, hence allowing any community of agents to automatically perform *what-if* analysis of potential new interaction rules.

Keywords. normative multiagent systems, game theory, rules, logic programming, Institutional Analysis and Development framework

1. Introduction

In the field of normative multiagent systems (norMAS), a great deal of work has been devoted to the study of *norms*, *rules* and other mechanisms to achieve coordination among autonomous agents [1,2,3]. In parallel, game theory has provided a powerful toolbox to model multiagent interactions of competitive, cooperative and hybrid nature. Very well established game theoretical solution concepts are prevalent across the Multiagent Systems literature (e.g. [4,5]). However, in game theory, the rules that configure the structure of the interaction become irrelevant once the formal model has been built, and are often expressed in non-systematic, plain natural language.

The fundamental contribution of this paper is a formal methodology for the *what-if* analysis of community rules through the game structures they generate. To do so, we define the syntax of the novel Action Situation Language (ASL), inspired by the Institutional Analysis and Development (IAD) framework. We formally define its semantics as an extensive-form game (EFG) and provide an engine to automatically build it from an ASL description, hence connecting the norMAS and game theory fields. The choice of EFGs as the ASL semantics is motivated by the availability of many reasoning schemes from the game theory literature. The application of these schemes to the resulting model completes the pipeline from a rule specification to an evaluation of the outcomes it promotes.

[1] Contact: {nmontes,nardine,sierra}@iiia.csic.es

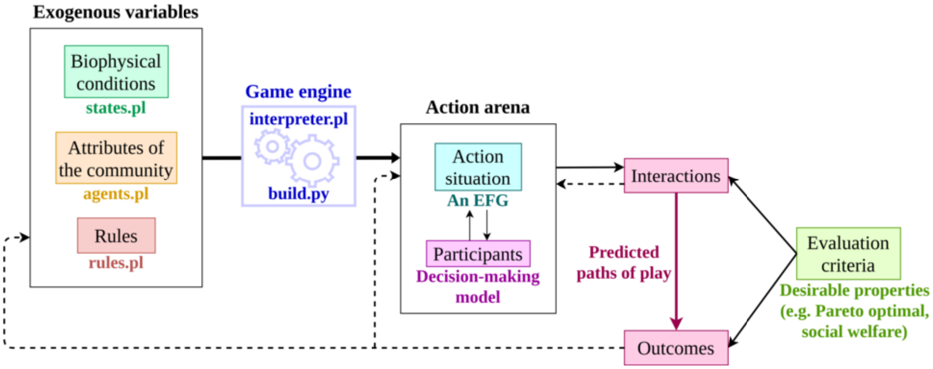


Figure 1. Outline of the IAD framework (adapted from [6, p. 15]). Coloured text outside boxes indicate either the script that contain the information on the boxed component, or the game-theoretical concepts that would represent it.

In the remainder of this paper, we provide some background on the IAD framework (Section 1.1) and review some research efforts similar to ours (Section 1.2). Then, we move on to the main part and present the syntax and semantics of ASL in Sections 2 and 3 respectively. We complement those two with a running example. Finally, we conclude and point to possible future directions in Section 4.

1.1. *Background*

Within the field of policy analysis, the Institutional Analysis and Development (IAD) framework, put forward by Ostrom and colleagues [6], represents a comprehensive theoretical effort to identify and delineate the universal building blocks that make up any social interaction. Its main components are presented in Figure 1. In the central part, the social interaction under study (e.g. a commercial transaction, a legislature) is referred to as an *action arena*. It is made up of a set of *participants* (endowed with some decision-making model) who find themselves within an *action situation*, which is broadly defined as the social space they might enter, take actions in and jointly bring about outcomes.

Action arenas are conditioned by three sets of exogenous variables: biophysical conditions, attributes of the community, and rules. The first two refer to environmental and material features. Within the scope of this work, the term *rules* will encapsulate both the laws of nature that inevitably play a part in determining outcomes, as well as malleable human-made regulations that constrain or provide alternative avenues for a course of action. Physical laws are distinct from biophysical conditions in the sense that laws control the dynamics of the environment (drop an object and it will land on the ground) while biophysical conditions refer to static elements (like land topology). Meanwhile, human-made regulative rules play an essential part when it comes to achieving more positive outcomes, as measured by the evaluation criteria of choice. Suitable changes to the regulatory rules have the potential to steer the interaction towards more desirable endings, by modifying the incentives that agents face.

Within an action situation, the IAD framework also delineates seven distinct internal variables that compose any social space. Of these, we will focus on the four that we deem indispensable: (1) the participants who are allowed to enter the interaction; (2) the roles they take on; (3) the actions assigned to every role; and (4) the linkages between the executed actions and the outcomes they bring about. Every one of these components will have a dedicated rule type in our language.

1.2. Related work

Originally, the IAD framework was complemented by the Institutional Grammar (IG) syntax [7], which parses institutional statements into several distinct fields. Lately, the IG has spurred renewed interest, with extensions to the original proposal including the nesting of statements [8] and the distinction between different levels of granularity in the parsing [9]. Although the early version of IG did include the derivation of some game-theoretical analysis from a set of statements [6, Ch. 5-6], no attempt is made to automate this process, as the IG syntax is not formulated as a *machine-readable* language. Some works that attempt to make it operational are limited in their scope [10,11], as they only use statements to encode agent strategies in an evolutionary type simulation.

On another front, the field of General Game Playing within the AI community has come up throughout the years with machine-processable languages for the specification of games. The most prominent of these is the Game Description Language (GDL) [12] and its extensions to imperfect-information [13] and epistemic games [14]. Beyond game playing, GDL has been used for more socially relevant applications, such as mediated dispute resolution [15] and automated negotiations [16].

The original GDL admits a form of restricted imperfect information as simultaneous moves that we incorporate into our language. However, the rules of the game are implicit in GDL descriptions, while our language represents them explicitly and individually (one by one). Therefore, ASL descriptions are more declarative than GDL ones, as new rules can be easily added and their individual impact examined.

Also within the AI community, declarative action representations are ubiquitous in the planning domain. The multiagent extension to the Planning Domain Definition Language (MA-PDDL) [17] and ASL have similar expressive power for actions: both allow the specification of concurrent actions with probabilistic effects. A difference between the two representations is the removal of state fluents: MA-PDDL specifies them explicitly within action effects, while ASL relies on incompatibilities between the previous state fluents and the newly derived ones to remove outdated facts.

2. ASL syntax

Now, we turn to the definition of the syntax of our Action Situation Language (ASL). To do so, we leverage the conceptual clarity of the IAD framework and tailor the design of our language to the components delineated in that theory. ASL is a logical language implemented in Prolog, hence fully machine-readable yet relatively syntactically friendly. In order to fully specify a complete action situation, it should, first, include constructs for the three sets of exogenous variables that determine it (see Figure 1):

- The candidate agents to take part in the interaction, plus any relevant characteristics (**attributes of the community**): age, gender, ethnicity, etc.
- The **biophysical and environmental conditions**, like land topology, location of resources, etc.

Table 1. Action Situation Language keywords, sorted into reserved predicate symbols (with their arity) and operators (with their type in parenthesis).

Predicates			Operators	
`agent/1`	`rule/4`	`initially/1`	`if` (prefix)	`then` (infix)
`participates/1`	`role/2`	`incompatible/2`	`where` (infix)	∼ (prefix)
`can/2`	`does/2`	`terminal/0`	`withProb` (infix)	`and` (infix)

- The **rules** structuring the interaction, in particular the following four rule *types*:
  - ∗ **Boundary rules** determine who is allowed to participate in the interaction.
  - ∗ **Position rules** assign (possibly several) roles to participants.
  - ∗ **Choice rules** establish the actions available to every role under the current circumstances.
  - ∗ **Control rules** relate (possibly joint) actions to the effects they have.

Additionally, the following is also necessary:

- The starting point of the interaction, and the conditions under which it halts.
- Which facts describing a state are compatible with one another and can be simultaneously true (e.g., an individual cannot be at two different places at the same time).

The predicates for ASL are gathered in Table 1. Most of these appear as part of `rule` arguments, and only `agent`, `initially`, `terminal` and `incompatible` are used as standalone predicates.

We start by reviewing the predicate symbols that do not appear within rules. First, `agent(A)` simply designates `A` as an individual susceptible of entering the action situation. Second, `initially(F)` indicates that fact `F` holds true at the start of the interaction, prior to any action being executed. `terminal` plays the opposite role, as it returns true whenever the conditions for halting the interaction are met. Finally, `incompatible(F,L)` states that fact `F` cannot be simultaneously true with the fluents in list `L`.

We move on now to the syntax of rules. All rule clauses, regardless of the component they target, follow the general template in Figure 2. Their first argument is an identifier for the action situation where they apply. The second argument is their type. Third, the priority is a non-negative integer that determines which rule is to prevail in case several clauses have contradicting effects. Rules that model the unregulated situation[2] are assigned priority equal to zero, and are referred to as the *default* rules. The overwriting operator ∼ is introduced to have high priority rules nullify the effects of lower priority rules. The fourth and last argument of a `rule` predicate contains its content expressed as an *if-then-where* construct. The content of the `Condition` and `Consequence`

[2] By “unregulated”, we mean that only rule statements that reflect physical principles are considered.

```
    Rule ::=  rule(Id,Type,Priority,
                  if Condition then Consequence where Constraints).
    Type ::=  boundary | position | choice | control
Priority ::=  0 | 1 | ... | ∞
```

Figure 2. General syntax of *if-then-where* rules.

Table 2. Syntactic restrictions for the Condition and Consequence fields for every of the proposed rule types. α stands for an atom, i.e. a predicate symbol with terms as arguments.

Rule type	Condition	Consequence
Boundary	agent(Ag)	[∼]participates(Ag)
Position	participates(Ag)	[∼]role(Ag,R)
Choice	role(Ag,R)	[∼]can(Ag,Ac)
Control	joint_action	[consequence$_1$ withProb p_1, consequence$_2$ withProb p_2, ...]
joint_action ::= does(Ag,Ac) [and joint_action]		
consequence ::= α [and consequence]		

fields is determined by the rule type in question. These restrictions are summarised in Table 2. Constraints always consists of a list of literals whose free variables unify with those in Condition and Consequence. The separation of rule pre-conditions into a short Condition and a Constraints field is not technically indispensable, but rather a stylistic choice to help keep the syntax concise.

Note that, in Table 2, boundary, position and choice rules have an analogous syntax: one agent, participates or role predicate as the Condition, and participates, role or can as the Consequence, respectively. In contrast, the control rules may have in their condition multiple does predicates concatenated by the and operator to reflect the execution of joint actions. Their consequences, instead of a single predicate, consists of a list where each of its members consists of predicates concatenated with and, and the whole conjunction is assigned some probability with the operator withProb. In order for a control rule to be valid, the probability distribution over the potential consequences must be well-defined, i.e. all p_i must fall in the range $[0,1]$ and must add up to unity.[3]

Fishers example (syntax) The best way to understand the syntax of ASL is to provide a complete example of an action situation description.[4] Here, we present the model of an open fishery, where two fishers compete for two fishing spots (it is assumed that one is more productive than the other) [18, Ch. 4].[5] The clauses are split into three files according to the three exogenous variables identified in the IAD framework. First, agents.pl contains the information on the attributes of the community. It declares two agents and two attributes for each, strength and speed (the second one will become relevant later on when we introduce higher priority rules).

Second, states.pl introduces the environmental conditions. Here, two fishing spots are declared. This file also contains the initially and terminal clauses. All fishers start at the shore. The interaction halts when both fishers are at distinct spots (first terminal clause) or when one of them has lost a fight (second terminal clause). The last piece of information in states.pl is the incompatible clauses. They indicate that a fisher can only be at one location at a time, and that only one of them may be the winner of a fight or race.

Third, rules.pl contains the rule base. It only contains the *default* rules with priority equal to zero. All of them use the identifier “fishers”. The first two rule statements are

[3] If that is not the case, the game engine (see Figure 1) will raise an error.

[4] Further examples can be found at the extended pre-print version of this paper, see: https://arxiv.org/abs/2105.13151.

[5] The complete ASL description for the fishers domain appears in Listing 6 (Appendix C) of the extended pre-print.

very generic boundary and position rules. They let all agents enter the action situation and denote all of them as fishers. Then, the choice rules indicate that fishers may go to any fishing spot from the shore and that, once at a spot, they may stay or leave for the other one.

The control rules regulate the effects that fishers' actions have on the environment. The first two control rules are related to the movement of fishers from the shore to a spot and between spots. Both of them are stated in terms of a single individual action and have deterministic effects (boats never break down). The last control rule does include joint actions and stochastic consequences. It states that when two different fishers who are at the same spot and take the same action will inevitably fight for the spot they meet at for the second time. The probability of each fisher winning the fight is proportional to their relative strength.

3. ASL semantics

As earlier introduced, an action situation description in ASL has its formal semantics grounded as an extensive-form game (EFG). This choice is motivated by the availability of well-established solution concepts within the game theory literature, which allow agents to reason on top of the resulting game. The construction of an EFG from an ASL description is composed of three main steps:[6] rule interpretation, game round building, and game round concatenation.

3.1. Rule interpretation

First, rule interpretation consists of querying the knowledge base to find, given the current state of the system, instantiations of the active rules (bindings to free variables), and processing their consequences. This task is performed by the interpreter.pl script (see Figure 1). Two groups of rules are distinguished regarding the processing of consequences.

On one hand, the common and relatively simple syntactic structure of boundary, position and choice rules make the processing of their consequences much easier. Essentially, what it amount to is the deletion of any fluent f if that same fluent preceded by the overwriting operator, $\sim f$, is derived from a higher priority rule. On the other hand, control rules need a much more thorough processing of consequences. Given a pre-transition state s_t (aka a set of fluents that completely characterise the current circumstances) and a joint action profile $\mu = \{\texttt{does}(ag_1, ac_1), \texttt{does}(ag_2, ac_2), ...\}$, the interpretation of control rules returns a set of potential post-transition states $S_{t+1} = \{s^1_{t+1}, s^2_{t+1}, ...\}$ and a probability distribution over those $P : S_{t+1} \to [0,1]$.

We do not go into the details of this derivation and refer to the extended report for a detailed exposition. However, it is worth explaining how the interpretation of control rules tackles the *frame problem* [19]. This is an issue that any action formalism has to address. It states that, when the effects of an action are axiomatised, it should only be necessary to state the facts that do change due to it. Listing all variables that are not affected by a particular action is not to be required. This is precisely the case in ASL,

[6] This paper presents only an overview of the game building process. For a more detailed report, see Sections 3 and 4 of the extended pre-print.

```
function BUILD-GAME-ROUND($s_t$):
Interpret choice rules to get the available action for every agent at $s_t$
Starting from the root node (identified with $s_t$), add an information set for every agent
For every terminal node $z$:
    Get the joint action profile $\mu$ executed to get from the root to $z$
    Interpret the control rules to get the potential next states $S_{t+1}$ and probabilities $P$
    If there are stochastic effects ($\mid S_{t+1} \mid > 1$):
        Turn $z$ into a chance node and add one child for every $s_{t+1} \in S_{t+1}$
        Set the edge probability to $P(s_{t+1})$
    Else identify $z$ with $s_{t+1}$ (the only element in $S_{t+1}$)
```

Algorithm 1. BUILD-GAME-ROUNDS constructs the EFG that represents the execution of one action for agent in state s_t.

as control rules state in their `Consequence` field only the terms that do change. Then, prior to returning the set of potential next states S_{t+1}, the rule interpreter goes through the fluents in the pre-transition state s_t and, by performing queries to the `incompatible` clauses, determines which facts may be carried over to the post-transition states. Hence, fluents that are not affected by the actions in μ remain part of the state description.

3.2. Game round building and concatenation

The construction of the action situation semantics is performed by the script build.py (see Figure 1), which repeatedly communicates with the rule interpreter to get the processed consequences of rules.[7] In order to build the complete EFG semantics of an ASL description, the process is divided into the consecutive constructions of *game rounds*:

Definition 1. A *game round* is an extensive-form game[8] with the following characteristics:

- The root node is never a chance node.
- There is, at most, one information set[9] per player.
- For any two nodes x_1, x_2 that belong to the same information set, the length of the path from the root to x_1 and from the root to x_2 must be equal.
- If node x is a chance node, then all of its children are terminal.

In practice, a game round is an EFG where every agent has the opportunity to make at most one move. With imperfect information, the moves by every player are modelled as simultaneous. In this work, we use game rounds to model all the ways by which the system may transition from state s_t (the root of the game round) to a post-transition state in S_{t+1} (the terminal nodes) by executing any of the actions available at s_t according to the choice rules. We choose to use imperfect-information EFGs instead of normal-form games (the benchmark models for joint actions) because, through the use of chance

[7] The communication between Prolog and Python is achieved thanks to the open-source PySwip package: `https://github.com/yuce/pyswip`.

[8] For a thorough definition of EFGs, see [20].

[9] In an extensive-form game, an information set is a subset of a player's decision nodes such that, at the time of making a move, the player only knows that the system is in one of the subset's nodes, but not specifically which one.

```
function BUILD-FULL-GAME:
Interpret the boundary rules to get the set of participants
Interpret the position rules to get their roles
Set $s_0$ to the set of derivable instanced from initially(F)
$Q \leftarrow$ QUEUE($s_0$)
While $Q$ is not empty:
    $s_t \leftarrow$ POP($Q$)
    If ?- terminal returns true at $s_t$ then continue
    $\gamma \leftarrow$ BUILD-GAME-ROUNDS($s_t$)
    Append $\gamma$ to overall game tree by $s_t$
    Push the terminal nodes in $\gamma$ to $Q$
```

Algorithm 2. BUILD-FULL-GAME constructs the complete EFG semantics of an ASL description by concatenating game rounds.

nodes, EFGs explicitly store the information on the stochastic dynamics of the environment, a feature that is not available in normal-form games. Pseudo-code for the construction of game rounds is shown in Algorithm 1.

Now that we know how to build a single round, the only step that is left is their concatenation in order to build the complete game. Prior to that, the boundary and position rules have to be interpreted to get the participants and their roles, and the initial state of the system is derived as the set of instantiations of `initially(F)`. The pseudo-code for the function that concatenates the game rounds and builds the complete EFG semantics appears in Algorithm 2.

Note that, by construction, some of the nodes in the final game tree cannot be identified with the actual states of the system, but are auxiliary nodes necessary to capture the simultaneity of moves. Similarly, chance nodes do not correspond to actual states, but are needed to explicitly store the probabilities of random effects. In fact, the only nodes that can be identified with an actual state (i.e., with a set of fluents that completely characterise the circumstances of the system) are the root nodes of game rounds and the terminal nodes.

Typically, extensive-form games have some numerical rewards assigned to every agent at their leaf nodes. These quantities, typically referred to as the *utilities* of the game, serve as the objective function to implement various reasoning schemes. Our game building algorithms, however, do not assign utilities to the resulting leaf nodes. Once the complete game tree is constructed, we leave it to the discretion of the user to set rewards *a posteriori* (e.g., as a function of the fluents that hold at the terminal nodes and/or the path of play from the root node).

Fishers example: Semantics We complement the fishers action situation description with its corresponding game semantics, which appear in Figure 10 of the extended pre-print version. This extensive game has been built solely from the default rules, intended to capture the dynamics of the unregulated situation.

To illustrate the addition of a new policy, we append some extra rules to the action situation description with priority 1. This new extended description constitutes the *first-in-time, first-in-right* configuration. The additional rules are displayed in Listing 7 in Appendix C of the extended pre-print. Now, when agents leave the shore for the same fishing spot, they race to get there. The winner of the race is determined by the same mechanism as the loser of the fight was (by flipping a biased coin), but with the `speed`

Table 3. Terminal node and its associated state fluents that are most likely to be reached under the two rule configurations for the fishers action situation.

Rule configuration	Most likely outcome (Probability)
Default	15 - `at(alice,spot1),at(bob,spot1),won_fight(alice)` (0.31)
First-in-time, first-in-right	13 - `at(alice,spot2),at(bob,spot1),won_race(bob)` (0.62)

of the agents instead of their `strength`. This is captured by the last rule of type control. Then, the winner of the race is guaranteed the spot, meaning that he is obliged to stay, while the loser must leave. These requirements correspond to the two first rules of type choice. The resulting game semantics appear in Figure 11 of the extended pre-print version.

We set the utilities to the resulting game trees by assigning the following benefits and costs to some of the actions and outcomes: a fisher keeps a spot to himself or wins the fight over it ($v_1 = 10, v_2 = 5$), a fisher looses a fight ($d = -6$), a fisher travels between spots ($c = -2$). Then, we implement the computation of subgame perfect equilibrium (SPE) strategies, by computing the Nash equilibria at the final game rounds (following [21, p. 104]) and backtracking the expected utilities. In fact, the game semantics of ASL are particularly well suited for the implementation of subgame perfection rationality schemes, as every subgame corresponds to a combination of game rounds, and these are typically much smaller in size than the overall game tree, hence reducing computation requirements.

The most likely terminal nodes, and their associated fluents, predicted by the SPE strategies are displayed in Table 3. The default rule configuration predicts violence in the most likely outcome, whose probability is around 30%. In fact, this rule configuration leads to a violent outcome (leaf nodes 14 through 17, and 24 through 27) around 50% of the times the game is played. In contrast, the implementation of *first-in-time, first-in-right* rules avoid violence. By this evaluation criteria (avoidance of violence), the additional rules certainly lead to a more socially desirable outcome, thus the community may collectively agree to incorporate them.

4. Conclusions

In this work, we have defined the syntax and semantics of the Action Situation Language, which turns descriptions of social interactions into formal game models that can be later examined using the standard tools of game theory. Our contribution, coupled with some model of individual rationality and an evaluation criteria for the potential outcomes, amounts to a complete computational model of Ostrom's IAD framework.

The most interesting use that can be made of ASL is as a tool for the formal *what-if* analysis of community rules. The ability to introduce and retract rules into a single description is a feature that sets ASL apart from other game-oriented logical languages. We have illustrated such an analysis with an example of interest for policy analysts, the regulation of an open fishery through the introduction of *first-in-time, first-in-right* rules.

The work presented here can be expanded into several directions. For example, formal aspects of ASL, such as its integration with an action formalism (e.g. Situation Calculus), could be explored. On the more practical side, refinements to the language can also help enhance its expressive power. For example, a new type of information rules, whose consequences deal with `sees` or `knows` predicates, could be introduced to regulate the observability of the current state, opening the door for extending the use of imperfect information beyond the modelling of simultaneous actions.

References

[1] Yoav Shoham and Moshe Tennenholtz. On social laws for artificial agent societies: off-line design. *Artificial Intelligence*, 73(1-2):231–252, 1995.

[2] Shmuel Onn and Moshe Tennenholtz. Determination of social laws for multi-agent mobilization. *Artificial Intelligence*, 95(1):155–167, aug 1997.

[3] Giulia Andrighetto, Guido Governatori, Pablo Noriega, and Leon van der Torre. Normative Multi-Agent Systems (Dagstuhl Seminar 12111). *Dagstuhl Reports*, 2(3):23–49, 2012.

[4] Carsten Hahn, Thomy Phan, Sebastian Feld, Christoph Roch, Fabian Ritz, Andreas Sedlmeier, Thomas Gabor, and Claudia Linnhoff-Popien. Nash equilibria in multi-agent swarms. In *Proceedings of the 12th International Conference on Agents and Artificial Intelligence*. SCITEPRESS - Science and Technology Publications, 2020.

[5] Philippe Caillou, Samir Aknine, and Suzanne Pinson. Searching pareto optimal solutions for the problem of forming and restructuring coalitions in multi-agent systems. *Group Decision and Negotiation*, 19(1):7–37, nov 2009.

[6] Elinor Ostrom. *Understanding Institutional Diversity*. Princeton University Press, September 2005.

[7] Sue E. S. Crawford and Elinor Ostrom. A grammar of institutions. *American Political Science Review*, 89(3):582–600, 1995.

[8] Christopher Frantz, Martin K. Purvis, Mariusz Nowostawski, and Bastin Tony Roy Savarimuthu. nADICO: A nested grammar of institutions. In *Lecture Notes in Computer Science*, pages 429–436. Springer Berlin Heidelberg, 2013.

[9] Christopher K. Frantz and Saba Siddiki. Institutional grammar 2.0: A specification for encoding and analyzing institutional design. *Public Administration*, 2021.

[10] Amineh Ghorbani and Giangiacomo Bravo. Managing the commons: a simple model of the emergence of institutions through collective action. *International Journal of the Commons*, 10(1):200–219, 2016.

[11] Alex Smajgl, Luis R. Izquierdo, and MArco Huigne. Modeling endogenous rule changes in an institutional context: the adico sequence. *Advances in Complex Systems*, 11(02):199–215, 2008.

[12] Michael Genesereth, Nathaniel Love, and Barney Pell. General game playing: Overview of the aaai competition. *AI Magazine*, 26:62–72, 06 2005.

[13] S. Schiffel and M. Thielscher. Representing and reasoning about the rules of general games with imperfect information. *Journal of Artificial Intelligence Research*, 49:171–206, 2014.

[14] Michael Thielscher. Gdl-iii: A proposal to extend the game description language to general epistemic games. *Frontiers in Artificial Intelligence and Applications*, 285:1630–1631, 2016.

[15] Dave de Jonge, Tomas Trescak, Carles Sierra, Simeon Simoff, and Ramon López de Mántaras. Using game description language for mediated dispute resolution. *AI & SOCIETY*, 34(4):767–784, 2017.

[16] Dave de Jonge and Dongmo Zhang. GDL as a unifying domain description language for declarative automated negotiation. *Autonomous Agents and Multi-Agent Systems*, 35(1), 2021.

[17] Daniel L. Kovacs. A multi-agent extension of pddl3.1. In *Proceedings of the 3rd Workshop on the International Planning Competition (IPC), 22nd International Conference on Automated Planning and Scheduling (ICAPS-2012)*, pages 19–27. ICAPS, 2012.

[18] Elinor Ostrom, Roy Gardner, and Jimmy Walker. *Rules, Games, and Common-Pool Resources*. University of Michigan Press, 1994.

[19] Fangzhen Lin. *Situation Calculus*, volume 3 of *Foundations of Artificial Intelligence*, chapter 16, pages 649–669. Elsevier, 2008.

[20] Julio Díaz. *An introductory course on mathematical game theory*. American Mathematical Society and Real Sociedad Matemática Española, Providence, Rhode Island, USA and Madrid, 2010.

[21] Yoav Shoham and Kevin Leyton-Brown. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, October 2014.

This page intentionally left blank

Sentiment Analysis and Text Analysis

This page intentionally left blank

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210122

Countering Negative Effects of Hate Speech in a Multi-Agent Society

Arthur MÜLLER[a,b] and Maite LOPEZ-SANCHEZ [a,1]
[a] *Universitat de Barcelona,* [b] *University of the Bundeswehr Munich*

Abstract. Hate speech expresses prejudice and discrimination based on personal characteristics such as race or gender. Research has proven that the amount of hateful messages increases on online social media. If not countered properly, the spread of hatred can overwhelm entire societies. This paper proposes a multi-agent model of the spread of hatred. We reuse insights from previous research to construct and validate a baseline model. From this, three countermeasures are modelled and simulated to investigate their effectiveness in containing the spread of hatred: (1) The long-term measure of education is very successful, but it still cannot eliminate hatred completely; (2) Deferring hateful content has a similar positive effect with the advantage of being a short-term countermeasure; (3) Extreme cyber activism against hatred can worsen the situation and even increase the likelihood of high polarisation within societies.

Keywords. hate speech, hate spread, countermeasures, social networks, opinion diffusion, education, deferring hate content, cyber activism.

1. Introduction

The United Nations define hate speech as the attack or usage of pejorative or discriminatory language with reference to a person or a group based on their religion, ethnicity, nationality, gender or other identity factor[2]. The usage of hateful language has become common on online social media. This is specially the case on platforms with slack content policies—such as Gab—, where the amount of hateful messages has steadily increased over the last years [15]. But also hateful users on platforms such as Twitter have become more extreme [9]. In fact, spread of their messages seems to be inadvertently supported by the algorithms of the social networks [17].

To counter this problem many measures with different temporal horizons have been proposed by researchers and politicians. The long-term effect of education is supposed to introduce positive bias into society by teaching democratic values and tolerance. Awareness campaigns focus on mid-term effects and try to prevent forming negative prejudices against out-groups and minorities. Others propose expensive manual community management or intrinsically motivated counter speech [8]. In contrast, the short-term measure of automatic filtering is criticised as the infringement of the freedom of speech. Exhaus-

[1] Research funded by projects SGR 341, MISMIS (PGC2018-096212-B-C33), Crowd4SDG (H2020-872944), CI-SUSTAIN (PID2019-104156GB-I00), COREDEM (H2020-785907), and nanoMOOC (COMRDI18-1-0010-02).

[2] https://www.un.org/en/genocideprevention/documents/UN Strategy and Plan of Action on Hate Speech 18 June SYNOPSIS.pdf

tive evaluations, however, are still lacking and it is difficult to assess the effectiveness of these initiatives and, much less, to compare them among each other. Therefore, the objective of this paper is to propose an agent-based model as a sandbox for the simulation and comparison of countermeasures against the spread of hatred. Three countermeasures with different temporal effects are selected for this approach: education, deferring hateful content and counter activism.

2. Related Work

This section introduces research describing hateful users and behaviours, mathematical models of opinion spread, and existing simulations in the context of hatred.

2.1. Characterising Hateful Users in Social Networks

Hateful users have been found to exhibit a very different profile when compared to other users. From the psychological point of view, they are energetic, talkative, and excitement-seeking [16]. However, other personality traits such as narcissism, lack of empathy, and manipulative are also attributed to them [7]. Haters show high activity on social media and follow more people per day. Although hateful users gain 50% less back-followers for every spawned following relationship per day, they can receive much more followers over the lifetime of their accounts due to their high activity [18]. Surprisingly, the amount of hateful persons is little and does not exceed 1% even on Gab, but they are responsible for an non-proportionally high amount of content. Furthermore, their content can spread faster and diffuse deeper into the network when forwarded by other users [14]. Hateful content seems to be less informative on Twitter, since less URLs and hashtags are added [18], but it is known to be more viral when enriched with images or videos [12]. Finally, hateful users are very densely connected and demonstrate higher reciprocity among themselves compared to normal users [18,14].

2.2. Models of Opinion Diffusion

A social network can be mathematically represented as a graph $G = (E,V)$, where users correspond to the set of vertices V and their relationships (i.e., interconnections) to the edges in E. Usually, individual opinions about a given topic are represented as numerical values in the interval $[0,1]$. Both limits of the interval are associated with the extreme stances about the considered topic.

Although opinions are iteratively formed by considering how users influence one another, different opinion diffusion models have been proposed in the literature. For example, the aim of DeGroot model [4] is to come to a consensus by using trust as a means to induce differences in the influence of users. DeGroot model was recently applied to the research on hateful behaviours on Twitter and Gab platforms to adjust the score for hate intensity of users [15,18]. In contrast, bounded confidence models follow the intuition that people usually do not accept opinions too far from their own, which is known as *confirmation bias*. For instance, Friedkin-Johnson [6] adds some kind of stubbornness, distinguishing between an intrinsic initial opinion, which remains the same, and an expressed opinion, which changes over time. Hegselmann-Krause (HK) [10] introduces confidence level—a threshold for opinion difference. Deffuant-Weisbuch (DW) [22] was the first to use asynchronous random opinion updates of two users considering the confidence level. Finally, Terizi et al. [21] conducted extensive simulations showing that HK and DW outperform other models in describing the spread of hateful content on Twitter.

2.3. Multi-Agent Simulations in the Context of Hatred and Polarisation

To the best of our knowledge, the majority of multi-agent simulations in the context of hatred and polarisation consider a two-dimensional grid as communication topology. Jager & Amblard [11] conducted a general simulation based on the Social Judgement Theory to demonstrate consensus and bi-polarisation. Stefanelli & Seidl [20] used the same theory to model opinion formation on a polarised political topic in Switzerland. The authors used empirical data to set up the simulation and validate their results. Bilewicz & Soral [3] proposed their own model of the spread of hatred. As apposed to this, Schieb & Preuß [19] employed the Elaboration Likelihood Model on a message-blackboard. In contrast to the models presented in Section 2.2, where the underlying psychological models use multiple influence factors to model opinion, these works rely on a simple combination of one-dimensional opinion values. In general, none of mentioned models considered more complex typologies of social networks, neither they studied countermeasures against the spread of hatred, which is the main contribution of this paper.

3. Terminology

Next, we introduce some terminology required to describe our models.

Hate score represents user's attitude and behaviour in terms of hatred. It is a real number in $[0,1]$, where both extremes correspond to a very non-hateful and hateful opinions respectively. We use the hate score as a user opinion value in our diffusion models. The same concept was also employed by Mathew et al. [15] who showed that hate score distribution on Gab is positively biased towards non-hateful stance. Similarly, we define a user as hateful when *hate score* ≥ 0.75, else as normal in accordance to and for better comparability with previous work. As stated before, the amount of haters is known to be a minority of ca. 1%. Therefore, we model the hate score using the Gamma distribution $\Gamma(\alpha,\lambda)$ as depicted in Figure 3, so that the area under curve for $x > 0.75$ is ca. 0.01. In rare cases when the Gamma distribution naturally exceeds the value of 1 we artificially set users' hate score to the extreme stance of 1.

Hate core is a network component consisting of densely connected hateful users as shown in Figure 1. Such components can be the result of high cohesiveness among hateful users and their higher activity. Although single users within a hate core do not exhibit the same influence as some famous mainstream users, as a compound they can achieve similar effects and attract other users. *Hate strains* can then emerge from a hate core as the result of opinion diffusion under negative influence as shown in Figure 2.

Swap to a hateful society takes place when the amount of hateful users exceeds 30% of all users in the social network. We consider such situation as the outcome of an irreversible process, which destabilises the society in a very severe way. Experiments have shown that after having trespassed this limit there is no return to a non-hateful society within the time scope of our simulation.

4. Baseline Model

Our baseline model is a multi-agent social network where users distribute content. The type of content depends on the user profile, which can be normal or hateful. Subsection 4.1 details how such users are added and connected in the network. Then, Subsection 4.2 reuses insights from the previous research in Section 2 to model the spread of hatred. Our aim is to ensure that the baseline model is close to reproduce findings from previous work. We name this model *baseline* because subsequent sections enrich it with different countermeasures and study their effectiveness in containing the spread of hatred.

4.1. Network Construction

The structure of a social network can be reproduced by the *preferential attachment* iterative method [2]. Briefly, in each round, when joining the network, new users connect to existing users with a probability corresponding to the *node degree* (amount of followers). So that users with many followers are more likely to receive new followers. Since this method does not distinguish between different user profiles, we extend it for hateful users. Firstly, we include hateful users according the Gamma distribution $\Gamma(10,25)$ in Figure 3 (see also Section 3). Secondly, we mimic their behaviour on Gab and Twitter as described in Subsection 2.1:

- When joining the network, a new hateful user will create twice new connections than a normal user. Specifically, we assign the connection variables $c_h = 2$ and $c_n = 1$, where h stands for hater and n for normal.
- A new hateful user will opt-in to connect to the group of hateful users with the probability $p_{h\rightarrow h} = 0.9$, else to normal users (the arrow → indicates connection). After that, the preferential attachment is applied to select a specific user within each group. A hateful followee[3] will answer with the same probability $p_{h\leftarrow h} = 0.9$ and spawn a following connection (the arrow ← indicates following back).
- Hateful users receive less followers from normal users per time interval. Hence, following back by normal users is modelled with $p_{n\leftarrow n} = 0.8$ and $p_{h\leftarrow n} = 0.4$. Lastly, haters will be less likely to follow back normal users with $p_{n\leftarrow h} = 0.08$ compared to the opposite direction.

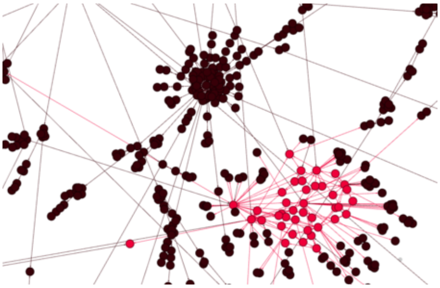


Figure 1. Hate core
Densely connected hateful users (red nodes)

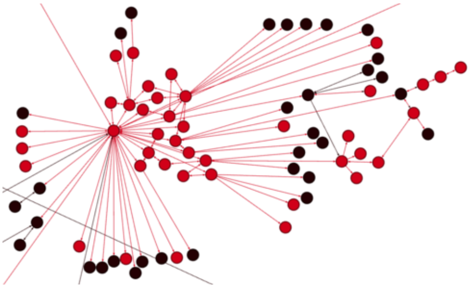


Figure 2. Hate strains
Hate core disseminating hatred in strains to nodes with lesser network density

[3] In the terminology of Twitter, a followee is a person who is followed by someone.

4.2. Opinion Diffusion by Content

In the bounded confidence models described in Section 2.2, the influence of users is limited to its followers. However, in social networks such as Twitter, the content created by users can be re-posted and, thus, arrive to and influence further audience. Here we reuse the concept of confidence level. Also, the formula of opinion adaption is borrowed from the DW model [22] (see Subsection 2.2), but applied to the author's opinion carried within a post. Thus, a post of user j will influence the opinion of user i by a factor $\mu = 0.05$ at the round k, if the difference of both opinions is below a confidence level τ_i:

$$x_{i,k} = x_{i,k-1} + \mu \cdot (x_{j,k-1} - x_{i,k-1}), \quad \text{iff} \quad |x_{i,k-1} - x_{j,k-1}| < \tau_i \tag{1}$$

where $x_{i,k}$ represents the opinion of user i at time k and the τ_i threshold is modelled as a triangular function on the users' opinion (hate score), so that extreme users will exhibit a rather fixed opinion.

When a post is created or re-posted by some user, then it can influence all of its followers as readers. Thereby:

- Hateful users will be more active and post at every round (i.e., with probability $p_{h\_pub} = 1$), whereas normal users only with $p_{n\_pub} = 0.2$.
- A post cannot be re-posted twice by the same user. However, it can be re-posted with some low probability even if the opinion does not correspond to re-poster's own opinion. We align here w.r.t. retweet statistics provided by Ribeiro et al. [18] and set re-posting probabilities between normal and hateful users to $r_{n\to n} = 0.15$, $r_{h\to h} = 0.45$, $r_{n\to h} = 0.05$ and $r_{h\to n} = 0.15$ (here, the arrow $\to$ indicates content flow). In this manner, a hater will re-post a normal post with the lowest probability.
- In order to consider different users' activity profiles, we limit the amount of re-posts by the same user per round by setting variables to $m_h = 6$ and $m_n = 2$.

5. Modelling Countermeasures

We enrich the baseline model with three alternative countermeasures aimed at containing the spread of hatred. Next subsections detail them.

5.1. Educational Bias

As mentioned in the introduction, one of the long-term effects of the education could be a positive bias introduced into the population. This can be modelled by skewing the distribution used for sampling of the *hate score* for society members (see Section 3). The mean value of the whole distribution should then move into the direction of non-hateful persons, hence decrease. However, we assume that despite the educational bias on the majority of the population, the group of very hateful persons will still be present in the population with the same proportion of ca. 1%. So, as depicted in Figure 3, we change the parameters of the Gamma distribution $\Gamma(\alpha, \lambda)$ so that their mean values μ are decreased. Hence, rather than modelling how this positive bias is actually introduced, we simply apply the positively-skewed distributions during the network construction phase (see Subsection 4.1).

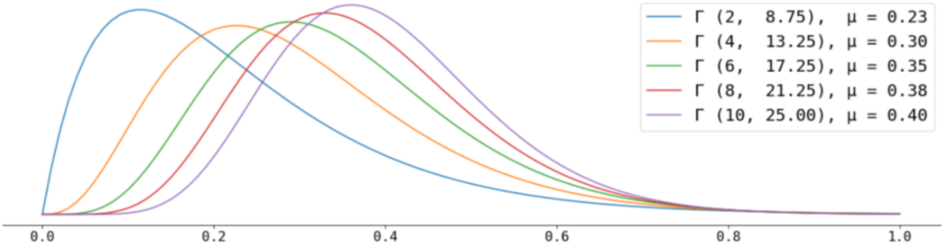


Figure 3. Alternative hate score probability distributions modelled with $\Gamma(\alpha,\lambda)$ distribution

5.2. Deferring Hateful Content

In contrast to the educational bias, deferring hateful posts is applied during the opinion dynamics phase described in Subsection 4.2. According to Dharmapala and McAdam [5], the perceived amount of hate speakers play a key role for motivating other users to join and engage in hate speech, making conversations more viral. It is also known that the lifetime of hateful conversations does not exceed a few days and has a culmination within the margins of one day [13]. Hence, the assumption is that deferring hateful content might decrease the willingness of responses. As far as we know, such reactions give more weight to the content and lead to better promotion of it by internal algorithms of social networks [17]. In this work, the response to such content is interpreted as re-posting. We employ a variable probability p_{defer} of deferring a post at some round. Any hateful post can be deferred again, if it is re-posted in further rounds. In addition to parameters in Section 4.2, a cumulative factor $f_{deferred} = 0.5$ is used to decrease the probability of being re-posted.

5.3. Counter Activism

The group of counter activists is aimed to be the pole of '*the good*'. They build a counter movement by convincing people to promote anti-hate slogans and to spread positively influencing messages. Such actors exhibit activity behaviours very similar to hateful users, but transport the opinions from the lower hate score interval $[0, 0.25)$. This countermeasure starts during the opinion dynamics phase, where activists are sampled from the group of non-hateful persons with a probability $p_{convince}$. By default, their opinion is not fixed and can change due to opinion diffusion. When it exceeds *hate score* ≥ 0.25 they change to normal activity, but keep previously created connections. Furthermore:

- On becoming activist (denoted as a), a person spawns additional following connections c_a to the group of all activists, which are answered with the probability $p_{a \leftarrow a} = 0.9$.
- Activists publish posts with the probability $p_{a\text{-}pub} = 1$ at every round.
- They never re-post any content of haters and vice versa, but promote non-hateful content frequently. For the rest, the rules of normal users are used. Therefore, the re-posting probabilities are $r_{a \to h} = r_{h \to a} = 0$, $r_{a \to a} = 0.45$ and $r_{a \to n} = r_{n \to a} = 0.15$.
- The maximal amount of re-posts per round is set to $m_a = 6$.

6. Simulation Results

The simulation is conducted in two phases. First, the network construction from Section 4.1 is run until $t_1 \in [0, 500, 1000, 2000, 5000]$ rounds, which creates a network with the same size of user nodes. Then, opinion dynamics from Section 4.2 starts so that the network is grown for further 1000 rounds. Each simulation is conducted 100 times for building the following average metrics:

- Fractions of normal or hateful users and posts.
- Mean and standard deviation of hate score distribution within the society.
- Ratio of network densities of hateful to normal users, which shows how much hateful users are more cohesive than normal users.
- Reciprocity of following within normal or hateful users.
- Mean amounts of followers and followees as well as follower-followee ratio.
- Mean path length of re-posts through the network.
- Fraction of swaps to a hateful society (see Section 3). Runs which end with a swap are not taken into account for none of the above metrics due to the instability they introduce. Instead, they are tracked separately through this specific metric.

6.1. Validating the Baseline Model

Validation of the baseline model is an important step for this work, since it normalises the simulation with real statistics on hateful users. In the first phase of simulation—network construction—multiple metrics could be satisfactorily reproduced in accordance to the state-of-the-art. However, runs resulted in extremely high network density ratios of hateful users over normal users: ca. 11 times more than reported by [14]. Also the amount of followers as well as the ratio between followers and followees of haters were to high compared to normal users. During the second phase—opinion dynamics—these metrics decreased very close to reported values for higher network sizes. Only the reciprocity among hateful users were too low compared to normal users. Although this might be repaired by introducing additional rewiring rules for users who switch from normal to hateful state, we advocate for the simplicity of the model and leave this for future work. Overall, it can be stated that our simulation represents hateful behaviours in a convenient way. Further, an interesting fact is that the switch from network growth to opinion diffusion demarcates a structural change in the sub-network of hateful users. It allows hate cores to disseminate hatred in strains to normal users with lesser network densities, showing that true hate cores might be even more densely connected than reported by statistics about real social networks.

6.2. Countermeasures simulation results

6.2.1. Educational Bias

We perform simulations of using education as a countermeasure by decreasing the α parameter of the Gamma distribution to induce stronger positive bias (see Figure 3 and Section 3). Overall, this countermeasure can be summarised as being very successful. On the one hand, it can substantially decrease the amount of hateful persons, even if not remove them completely from the society as can be seen on the left of Figure 4. Even with

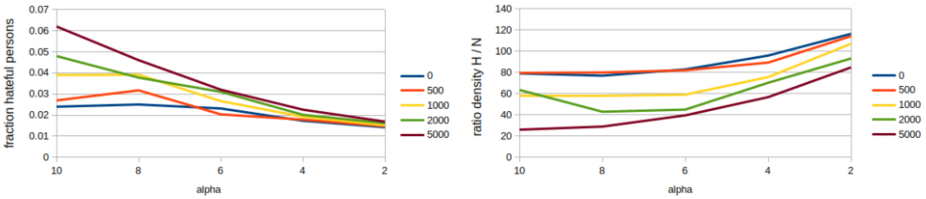


Figure 4. Effects of the education as countermeasure for different network sizes referred by the value of t_1: (left) fraction of hateful users; (right) ratio of network densities of hateful to normal users depending on the parameter α of the Gamma distribution

the strongest educational bias using $\alpha = 2$ the amount of haters does not fall below of 1%. Also, the amount of hateful posts decreases similarly. The risk of swaps to a hateful society drops from 25% below 5% for the value of $\alpha = 6$. The final mean hate score approaches even lower values than originally introduced by the Gamma distribution. This can be explained by the structure of the network produced using preferential attachment, where some nodes have unproportionally higher influence. Hence, applying a skewed distribution upon it can skew the final distribution even more after opinion diffusion.

On the other hand, the density among hateful users increases as depicted on the right of Figure 4. The same happens for the reciprocity and mean follower-followee ratio. Education impedes the emergence of hate strains, so that hateful persons stay among like-minded within highly densely connected hate cores. Surprisingly, the mean path length of hateful posts increases linearly. This is so because, although hate posts have much less room to unfold by re-posting within hate strains (see Figure 2), hateful posts can still make very long paths by circulating posts between persons within a hate core (see Figure 1).

6.2.2. *Deferring Hateful Content*

We conduct simulations by varying the deferring probability of hateful content p_{defer} and a value of 0.7 deems realistic considering the state-of-the-art accuracy in recognition of hate speech [1]. Compared to the education, this countermeasure is less successful in decreasing the fraction of hateful persons as can be seen on the left of Figure 5. There is even some kind of reluctance and increase for $p_{defer} < 0.5$. Something similar happens with the fraction of hateful posts and mean hate score. The reason for such reluctance, which can bear hidden risks especially in case of bad hate recognition accuracy, is not answered in this work and kept for future research. However, this countermeasure has an obvious effect in decreasing the mean path length of hateful content. More outstanding is its property in protection against swaps to a hateful society as shown on the right of Figure 4. This is very similar to the education, but provides a short-term effect. More importantly, it has the advantage of being much aligned with the freedom of speech value than the short-term countermeasure of automatic filtering.

6.2.3. *Counter Activism*

In the case of counter activists, we used different simulation setups with the four parameters in Table 1 with the aim of increasing the strength of the counter movement. Surprisingly, none of those simulations lead to a clear decrease of hateful users as depicted on the left of Figure 6. A decrease could be only recorded for settings with bigger networks

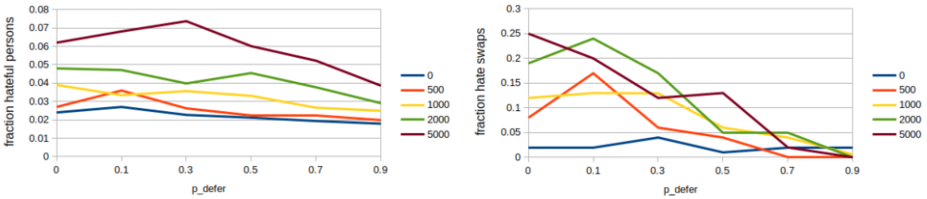


Figure 5. Effects of deferring hateful content as countermeasure for different network sizes t_1: (left) fraction of hateful users; (right) fraction of swaps to a hateful society depending on the deferring probability p_{defer}

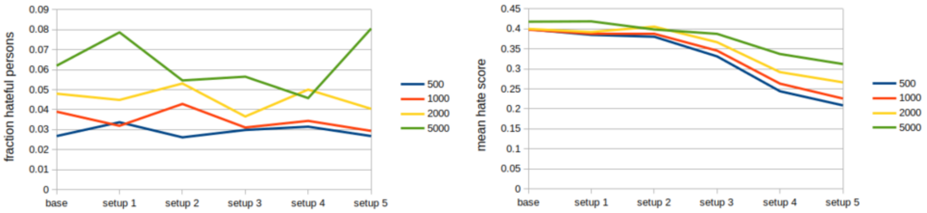


Figure 6. Effects of counter activists as countermeasure for different network sizes t_1: (left) fraction of hateful users; (right) mean hate score depending on the increasing strength of activist in subsequent setups

	setup 1	setup 2	setup 3	setup 4	setup 5
Convincing probability $p_{convince}$	0.01	0.01	0.04	0.01	0.01
Additional connections to other activists c_a	1	2	1	2	2
Fixed opinion (stubbornness)	false	false	false	true	true
Select activists by their influence	false	false	false	false	true

Table 1. Experiment settings for activists' countermeasure

over 5000 users in setups 2–4. The same happens to the fraction of hateful posts and, even more alarming, the fraction of swaps to a hateful society. Even so, a drop of the mean hate score was recorded—especially for the settings with stubborn activists—as seen on the right of Figure 6. Thus, activists seem to create higher polarisation within the society by dragging some persons into the positive direction without affecting hateful persons. This depletes representatives of the median opinion, so that people with higher hate scores are rather attracted by very hateful users. It shows that activism needs to be carried out in a very sensible way, which is beyond the scope of this paper.

7. Conclusions and Future Work

We propose a multi-agent model of the spread of hatred. We reuse insights from previous research to construct and validate a baseline model. Then, we enrich it by adding three countermeasures to study their effectiveness in containing the spread of hatred. As a result, we conclude that: i) The long-term measure of education is very successful, but it still cannot eliminate hatred completely; ii) Deferring hateful content has a similar positive effect with the advantage of being a short-term; iii) Extreme positive cyber activism can increase the society polarisation. Beyond that, we indicate that true hate cores, which are responsible for hate spread, might be even more densely connected than reported by statistics about real social networks. As future work, we plan to further refine the model (as mentioned along the paper) as well as to incorporate additional countermeasures.

References

[1] Sai Saketh Aluru, Binny Mathew, Punyajoy Saha, and Animesh Mukherjee. Deep learning models for multilingual hate speech detection, 2020.

[2] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[3] Michał Bilewicz and Wiktor Soral. Hate speech epidemic. the dynamic effects of derogatory language on intergroup relations and political radicalization. *Political Psychology*, 41:3–33, 2020.

[4] Morris H DeGroot. Reaching a consensus. *Journal of the American Statistical Association*, 69(345):118–121, 1974.

[5] Dhammika Dharmapala and Richard H McAdams. Words that kill? an economic model of the influence of speech on behavior (with particular reference to hate speech). *The Journal of Legal Studies*, 34(1):93–136, 2005.

[6] Noah E Friedkin and Eugene C Johnsen. Social influence and opinions. *Journal of Mathematical Sociology*, 15(3-4):193–206, 1990.

[7] Lena Frischlich, Tim Schatto-Eckrodt, Svenja Boberg, and Florian Wintterlin. Roots of incivility: How personality, media use, and online experiences shape uncivil participation. *Media and Communication*, 9(1):195–208, 2021.

[8] Iginio Gagliardone, Danit Gal, Thiago Alves, and Gabriela Martinez. *Countering online hate speech*. Unesco Publishing, 2015.

[9] Joshua Garland, Keyan Ghazi-Zahedi, Jean-Gabriel Young, Laurent Hébert-Dufresne, and Mirta Galesic. Countering hate on social media: Large scale classification of hate and counter speech. *arXiv preprint arXiv:2006.01974*, 2020.

[10] Rainer Hegselmann, Ulrich Krause, et al. Opinion dynamics and bounded confidence models, analysis, and simulation. *Journal of artificial societies and social simulation*, 5(3), 2002.

[11] Wander Jager and Frédéric Amblard. Uniformity, bipolarization and pluriformity captured as generic stylized behavior with an agent-based simulation model of attitude change. *Computational & Mathematical Organization Theory*, 10(4):295–303, 2005.

[12] Chen Ling, Ihab AbuHilal, Jeremy Blackburn, Emiliano De Cristofaro, Savvas Zannettou, and Gianluca Stringhini. Dissecting the meme magic: Understanding indicators of virality in image memes. *arXiv preprint arXiv:2101.06535*, 2021.

[13] Ping Liu, Joshua Guberman, Libby Hemphill, and Aron Culotta. Forecasting the presence and intensity of hostility on instagram using linguistic and social features. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 12, 2018.

[14] Binny Mathew, Ritam Dutt, Pawan Goyal, and Animesh Mukherjee. Spread of hate speech in online social media. In *Proceedings of the 10th ACM conference on web science*, pages 173–182, 2019.

[15] Binny Mathew, Anurag Illendula, Punyajoy Saha, Soumya Sarkar, Pawan Goyal, and Animesh Mukherjee. Hate begets hate: A temporal study of hate speech. *Proceedings of the ACM on Human-Computer Interaction*, 4(CSCW2):1–24, 2020.

[16] Binny Mathew, Navish Kumar, Pawan Goyal, and Animesh Mukherjee. Interaction dynamics between hate and counter users on twitter. In *Proceedings of the 7th ACM IKDD CoDS and 25th COMAD*, pages 116–124. 2020.

[17] Derek O'Callaghan, Derek Greene, Maura Conway, Joe Carthy, and Pádraig Cunningham. Down the (white) rabbit hole: The extreme right and online recommender systems. *Social Science Computer Review*, 33(4):459–478, 2015.

[18] Manoel Ribeiro, Pedro Calais, Yuri Santos, Virgílio Almeida, and Wagner Meira Jr. Characterizing and detecting hateful users on twitter. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 12, 2018.

[19] Schieb, Carla and Preuss, Mike. Considering the elaboration likelihood model for simulating hate and counter speech on facebook. *SCM Studies in Communication and Media*, 7(4):580–606, 2018.

[20] Annalisa Stefanelli and Roman Seidl. Opinions on contested energy infrastructures: An empirically based simulation approach. *Journal of environmental psychology*, 52:204–217, 2017.

[21] Chrysoula Terizi, Despoina Chatzakou, Evaggelia Pitoura, Panayiotis Tsaparas, and Nicolas Kourtellis. Angry birds flock together: Aggression propagation on social media. *arXiv preprint arXiv:2002.10131*, 2020.

[22] Gerard Weisbuch. Bounded confidence and social networks. *The European Physical Journal B*, 38(2):339–343, 2004.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210123

Discovering Dominant Users' Opinions in Reddit

Teresa ALSINET [1], Josep ARGELICH, Ramón BÉJAR and Santi MARTÍNEZ
INSPIRES Research Center – University of Lleida
Jaume II, 69 – 25001 Lleida, SPAIN

Abstract

Reddit is a social news aggregation and discussion website. Users submit content to the site such as links to news, which are then voted up or down by other members who in turn, can comment on others' posts to continue the conversation. In this work, we are interested in modeling how users interact with each other in Reddit debates, to discover the most dominant opinions in a debate. To this end, we introduce a user-based model for analysis of Reddit debates. In this model, comments by users are grouped per user, describing their opinion in relation to the root comment of the debate, and users are represented with a single node in a weighted graph, where node's weights represent relevance of user's opinions and edges represent agreement or disagreement relationships between users throughout the debate. In this model, agreement or disagreement between the opinions of two users is defined by aggregating the set of single interactions that have occurred between them during the debate. In this work we present a skeptical aggregation model for this task. For measuring the relevance of user's opinions, we consider two models: one based on the score of all the user's comments and other based on the user's karma, as computed by the Reddit platform. We characterize the set of most dominant opinions with an argumentative-based model, using the information of disagreement between opinions and relevance of opinions.

Keywords. Reddit, user-based model, skeptical aggregation, argumentation-based reasoning, consensus analysis.

1. Introduction

Reddit (http://www.reddit.com/) is a social news aggregation, web content rating, and discussion website. Users submit content to the site such as links, text posts, and images, which are then voted up or down by other members who in turn, can comment on others' posts to continue the conversation. This online debating platform is widely used to create long and deep debates with comments and answers to comments, where, thanks to the almost unlimited text length of Reddit comments (40,000 characters), users can express their opinions more accurately compared to other online debating platforms such as Twitter that restricts the number of characters to 280.

[1] Correspondence to: T. Alsinet. INSPIRES Research Centre, University of Lleida. C/Jaume II, 69. Lleida, Spain. Tel.: +34 973702734; E-mail: teresa.alsinet@udl.cat.

The analysis of opinions on general and specialized social networks, has received a lot of attention on many application fields. For example, there is a vivid interest in the analysis of tourists' opinions [13,16,17], and similar efforts are being done on marketing [5,9,7]. In order to understand what are the major accepted and rejected comments in different domains by Reddit users, in a previous work [3], we have proposed a representation model for analysis of Reddit debates oriented to comments. The model considers that debates are two-sided, where some of the comments are in agreement with the root comment of the debate, and the rest of the comments are in disagreement. Then, debates are represented as bipartite graphs where edges indicate disagreement between comments of these two disjoint sets.

Our aim in this work is to progress in the analysis of the debates in Reddit that allow us to identify structural relations between users' opinions and to be able to discover the most dominant users' opinions in a debate, that is, the opinions that are most widely accepted among the different users. To be precise, we propose to model how different users interact with each other in Reddit debates and to perform an analysis of dominant users' opinions based on argumentation frameworks. To this end, we introduce a natural extension of our previous comment-oriented model [3], that we call *user-based model* for Reddit debates. In this new model, comments within a debate are grouped by users or authors, such that comments of the same users that describe their opinion along the debate are represented by single nodes in the graph, and edges stand for agreement or disagreement relationships between users' opinions in the debate. When we represent a debate grouping comments by users, interactions between different users can give rise to circular agreement and disagreement relationships, and the agreement or disagreement of a user in relation to the opinion of another user in the debate should be defined by aggregating the set of single interactions that have occurred between them during the debate. In this work, the aggregation follows a skeptical approach giving rise to neutral interactions when a user is simultaneously in agreement with one part of an other user's opinion an in disagreement with the rest, which is a key difference with the approach we defined in our past work on a user-based model for Twitter discussions [2] where support relationships between users were not considered.

The final goal of the new user-based model is to find a set of users' opinions that is consistent and is widely accepted by most of the users, following acceptance semantics that come from argumentation theory, in which consistency is defined as absence of disagreement from one accepted opinion to another one, when the first opinion is considered more relevant than the second one. So, we characterize the set of most dominant users' opinions as this set of widely accepted and consistent user's opinions, where we use ideal semantics [8] to define this set. To the best of our knowledge, this is the first work to use an argumentation approach to analyze the dominant users' opinions in Reddit debates. Previous work in other social networks, like Twitter, has considered also the study of interactions between users, mainly using the *Twitter retweet graph* [15,14], but these works do not consider the aggregation of user's comments. For Reddit, studies are centered towards analyzing different dynamic characteristics of the discussions [6], or like in our past work on characterizing user profiles [3], but always using a comment-oriented model.

Argumentation has also been explored as a cybersecurity tool [12] for helping making decisions where a balance has to be found between security, operability and cost. The main difference between this field and our framework is that arguments are not comments on a social network, but statements about system configuration, sensor data, etc. And, as in many others fields, there is some degree of conflicting information with different possible causes and solutions for a specific problem. It has been noted [11], however, that argumentation-based decisions can be manipulated by attacking key arguments to overturn the solution. In the same way, we are currently studying how to extend the user-based model to detect users that have deliberately manipulated a debate by means of their *influence degree* [3] and, possibly, other measures.

The rest of the paper is organized as follows. In Section 2, we recall [3] the comment-based model for Reddit debates. Then, in Section 3, we formalize the user-based model to analyze the global behavior of users in debates based on a skeptical sentiment analysis approach to assess the agreement and disagreement relationship between users. Finally, in Section 4, we define and test our argumentation-based reasoning system to compute the set of dominant user's opinions for a Reddit debate.

2. Comment-based model for Reddit

In this section, we present the computational structure we have already defined [3] to analyze the agreement between comments in Reddit debates. This approach considers that debates are two-sided, where some of the comments are in agreement with the root comment of the debate, and the rest of the comments are in disagreement. We reference this debate structure as *comment-based model* for a Reddit debate. Next, we first formalize the notions of comment and Reddit debate for a root comment.

A *comment* c is a tuple $c = (m, u, sc)$, where m is the text of the comment, u is the user's identifier of the comment, and $sc \in \mathbb{Z}$ is the score of the comment.

Let $c_1 = (m_1, u_1, sc_1)$ and $c_2 = (m_2, u_2, sc_2)$ be two comments. We say that c_1 *answers* c_2 if c_1 is a reply to the comment c_2.

Let $r = (m_r, u_r, sc_r)$ be a comment such that m_r contains a link to some news. A *Reddit debate* on the root comment r is a non-empty set Γ of Reddit comments such that $r \in \Gamma$ and every comment $c \in \Gamma$, $c \neq r$, c answers a previous comment in Γ.

In Reddit, except for the root comment, each comment answers exactly one previous comment, usually by another user or author, and the score of a comment is computed as the sum of positive and negative votes that the comment has received. So, the score of a comment is negative whenever the comment has more negative votes than positive ones and positive, otherwise.

Given the structure of a Reddit debate Γ on a root comment r, the next step is to extract the relationships between the comments in Γ. We represent Γ as a labeled tree, where each comment gives rise to a node, edges denote answers between comments and are labeled with a value in the real interval $[-2, 2]$. The label for an edge (c_1, c_2) denotes the *sentiment* expressed in the text of the comment c_1 in response to the text of the comment c_2, so that, the value -2 denotes a total disagreement and the value 2 a total agreement. We use the sentiment value 0 to denote both answers expressing a neutral position with respect the opinion expressed in c_2, and answers expressing at the same time agreement with part of the opinion expressed in c_2, and disagreement with another part of c_2.

Formally, let Γ be a *Reddit debate* on a root comment r. A *Debate Tree* (DebT) for Γ is a tuple $\mathcal{T} = \langle C, r, E, L \rangle$ such that: (i) for every comment in Γ there is a node in C,

(ii) node $r \in C$ is the root node of $\mathcal{T}$, (iii) if c_1 answers c_2 then there is a directed edge (c_1, c_2) in E, (iv) L is a labeling function $L : E \to [-2, 2]$, where the value assigned to an edge denotes the sentiment of the answer, from highly negative (-2) to highly positive (2), and (v) only the nodes and edges obtained by applying this process belong to C and E, respectively.

Given a Reddit debate, we make its corresponding DebT using the Python Reddit API Wrapper (PRAW) [1] to download its set of comments, and then we evaluate the sentiment for an edge (c_1, c_2) in the DebT by means of the sentiment analysis software of [10] using the text of the comment c_1, where the value assigned denotes the sentiment of the answer, from highly negative (-2) to highly positive (2).

3. User-based model for Reddit

Our goal in this work is to introduce and investigate a suitable user-based model that allows us to represent how different users interact with each other in Reddit debates allowing us to discover the most dominant users' opinions. To this end, we group comments of a debate by user and we consider that the relationship between users' opinions of two users are defined from the agreement and disagreement relationships between the individual comments of these two users.

We consider debates in which every user's opinion is consistent (users do not contradict themselves) and where users are not self-referenced. That is, for each user u and each pair of comments $c_1 = (m_1, u, sc_1)$ and $c_2 = (m_2, u, sc_2)$, we assume that messages m_1 and m_2 do not express neither conflicting nor inconsistent information, and that c_1 does not respond to c_2, nor c_2 to c_1.

Given a debate Γ on a root comment r with users' identifiers $\{u_1, \ldots, u_n\}$, we define the *opinion* of the user $u_a \in \{u_1, \ldots, u_n\}$, denoted C_a, as the set of comments of u_a in Γ except for the root comment r. If we consider the particular case of root users (the users who post the root comment), we notice that, when they do not participate through the debate, their opinion is empty, denoting that they have only posted the (root) news while staying passive throughout the debate. This is intentional, since the root comment plays a special role in the debate, setting its topic. Thus, in order to be considered a "true participant" on the debate, the root user should contribute during the discussion. Notice that the Reddit platform itself distinguishes between root and non-root comments, since it provides two different global user metrics, Post Karma and Comment Karma, where the first one corresponds to the points achieved by posting interesting news (root comments) and the second one corresponds to the points achieved from non-root comments (debate generated on some root comment).

Next we formalize the graph we propose to represent user-centered debates, called *User Debate Graph*, where the nodes are the users of the debate denoting their opinion with respect to the root comment and the edges denote interactions between users. In addition, we associate each edge of the graph with a weight representing the overall sentiment of the interactions between users.

Definition 1 *(User Debate Graph) Let* Γ *be a Reddit debate on a root comment* r *with users' identifiers* $\{u_1, \ldots, u_n\}$ *and let* $\mathcal{T} = \langle C, r, E, L\rangle$ *be a* DebT *for* Γ*. A* User Debate Graph (UDebG) *for* $\mathcal{T}$ *is a tuple* $\mathcal{G} = \langle \mathcal{C}, \mathcal{E}, \mathcal{L}\rangle$*, where*

[1] https://github.com/praw-dev/praw

- *$\mathcal{C}$ is the set of nodes of $\mathcal{G}$ defined as the set of users' opinions $\{C_1, \ldots, C_n\}$; i.e. $\mathcal{C} = \{C_1, \ldots, C_n\}$ with $C_a = \{(m, u_a, sc) \in \Gamma \mid (m, u_a, sc) \neq r\}$, for all users $u_a \in \{u_1, \ldots, u_n\}$.*
- *$\mathcal{E}$ is the set of edges of $\mathcal{G}$ defined as the set of interactions between different users in the debate; i.e. there is an edge $(C_a, C_b) \in \mathcal{E}$, with $u_a, u_b \in \{u_1, \ldots, u_n\}$ and $u_a \neq u_b$, iff there is $(c_1, c_2) \in E$ such that $c_1 \in C_a$ and $c_2 \in C_b$.*
- *$\mathcal{L}$ is a sentiment weighting scheme for $\mathcal{E}$; i.e. a map $\mathcal{L} : \mathcal{E} \to [-2, 2]$ assigning to every edge $(C_a, C_b) \in \mathcal{E}$ a value in $[-2, 2]$, that expresses the overall sentiment of the user u_a regarding the comments of the user u_b in the debate, from highly negative (-2) to highly positive (2). For each edge $(C_a, C_b) \in \mathcal{E}$, the value $\mathcal{L}(C_a, C_b)$ is meant to be computed from the individual sentiment of each answer in C_a to a comment in C_b; i.e. from the set $\{L(c_1, c_2) \mid (c_1, c_2) \in E,\ c_1 \in C_a \text{ and } c_2 \in C_b\}$.*

Only the nodes and edges obtained by applying this process belong to $\mathcal{C}$ and $\mathcal{E}$, respectively.

Notice that if a user only responds to the news of the debate (the root comment r), the user is mapped in the UDebG to a node in $\mathcal{C}$ with zero output degree denoting that the user starts no discussion with other users. In addition, users whose comments do not generate answers from other users are represented with nodes whose input degree is zero. Therefore, isolated nodes may appear in the UDebG that correspond to users who have neither generated nor participated in the debate, that is, users that have only answered to the news and whose opinions can be considered as accepted by all users since they have not been discussed yet.

In the UDebG, each node denotes a user's opinion and relationships between nodes are mined from the prevailing sentiment among the aggregated comments of those nodes. Moreover, a user can answer comments of different users in a debate, and thus, can agree or disagree with several users. However, if a user u_a answers several comments of a same user u_b, the set of interactions between them is represented with a single edge (C_a, C_b) in $\mathcal{E}$ and with a single sentiment value $\mathcal{L}(C_a, C_b)$, which is meant to denote the overall sentiment of agreement or disagreement of the user u_a with respect to the user u_b.

Analogously, as we did in the previous section to define the sentiment relation between comments, we propose here to use a *skeptical sentiment scheme* for weighting the agreement or disagreement relation between users. Our skeptical approach is based on stating that a user agrees or disagrees with another user if and only if one can be completely sure of it.

Our aim is to define a sentiment weighting scheme $\mathcal{L} : \mathcal{E} \to [-2, 2]$ for edges in $\mathcal{E}$, by combining, in a skeptical way, the individual sentiment values assigned to the responses between comments; i.e. for any pair of users u_a and u_b with opinions C_a and C_b, respectively, and with $(C_a, C_b) \in \mathcal{E}$, $\mathcal{L}(C_a, C_b)$ is defined from the values in $\{L(c_1, c_2) \mid c_1 \in C_a \text{ and } c_2 \in C_b\}$. To be precise, we define the skeptical sentiment relation of the opinion C_a of the user u_a with respect to the opinion C_b of the user u_b as follows:

- u_a agrees with u_b, denoted as $C_a \to C_b$, iff all answers from the user u_a to the user u_b are positive, i.e. $C_a \to C_b$ iff $L(c_1, c_2) > 0$, for all $(c_1, c_2) \in E$ with $c_1 \in C_a$ and $c_2 \in C_b$;

- u_a disagrees with u_b, denoted as $C_a \rightsquigarrow C_b$, iff all answers from the user u_a to the user u_b are negative, i.e. $C_a \rightsquigarrow C_b$ iff $L(c_1, c_2) < 0$, for all $(c_1, c_2) \in E$ with $c_1 \in C_a$ and $c_2 \in C_b$; and
- u_a is neutral with respect to u_b, otherwise.

The neutral relation between users denotes both interactions expressing an overall neutral position of the user u_a with respect to the comments of the user u_b, and interactions expressing at the same time agreement with some of the comments in C_b, and disagreement with the rest. Moreover, as we are assuming a skeptical scheme, for both agreement and disagreement relationships between users we aggregate the overall sentiment as the minimum value of the individual answers which corresponds to a pessimistic interpretation of the degree of agreement and disagreement among users. This aggregation model tries to represent the fact that in a debate with multiple negative responses from one user to another, a pessimistic analysis will focus on the most negative response, and in the case of multiple positive responses, on the softest positive response.

Formally, we define the *skeptical sentiment weighting scheme* $\mathcal{L} : \mathcal{E} \to [-2, 2]$ for edges $(C_a, C_b) \in \mathcal{E}$ based on a *pessimistic* aggregation model of the degree of agreement or disagreement among users' comments as follows:

$$\mathcal{L}(C_a, C_b) = \begin{cases} \min_{\{(c_1,c_2)\in E \mid c_1 \in C_a \text{ and } c_2 \in C_b\}} L(c_1, c_2), & \text{if } C_a \to C_b \text{ or } C_a \rightsquigarrow C_b \\ 0, & \text{otherwise} \end{cases}$$

Notice that for agreement relationships ($C_a \to C_b$) we consider the least agreement value (i.e. the value closest to 0), while for disagreement relationships ($C_a \rightsquigarrow C_b$) we consider the highest disagreement value (i.e. the value closest to -2). We use the neutral sentiment value 0 for neutral relations.

The UDebG for a given DebT may contain cycles. These cycles provide fundamental information about the different relationships that are established from the interactions between different users. In this sense, aggregating the information by users allows us to identify the set of users whose opinions are accepted by most users, the authors involved in a circular argumentative discussion, or the most controversial users.

4. Discovering dominant users' opinions

Once we have introduced the user-based model of debates in Reddit, the next step is to characterize the set of the most dominant user's opinions among the users' opinions of the debate. To this end, we extend the argumentation-based reasoning system we have already developed [1] to analyze discussions in Twitter, to deal here with Reddit debates to find a set of users' opinions that is consistent and is widely accepted by most of the users, following acceptance semantics that come from argumentation theory, in which consistency is defined as absence of disagreement from one accepted opinion to another one, when the first opinion is considered more relevant than the second one. The approach, described in the rest of the section, consists of mapping a UDebG graph for a Reddit debate to a value-based abstract argumentation framework and to use ideal semantics to define the most dominant users' opinions.

4.1. The argumentation-based reasoning system

Given a UDebG with a sentiment scheme $\mathcal{L}$ for weighting the agreement or disagreement relationship between users, we build a corresponding value-based argumentation framework where arguments represent users' opinion and attacks between arguments represent disagreement relationships between users. Value-based abstract argumentation, introduced by Bench-Capon [4], is an extension of abstract argumentation with a valuation function *Val* for arguments taking values on a set R equipped with a (possibly partial) preference relation *Valpref*.

Definition 2 *(VAF for a* UDebG*) Let Γ be a Reddit debate on a root comment r with users identifiers $\{u_1, \dots, u_n\}$ and let $\mathcal{G} = \langle \mathcal{C}, \mathcal{E}, \mathcal{L} \rangle$ be a* UDebG *for Γ. A Value-based Argumentation Framework (VAF) for $\mathcal{G}$ is a tuple $\mathcal{F} = \langle \mathcal{C}, attacks, R, Val, Valpref \rangle$, where*

- *each node or user's opinion in $\mathcal{C} = \{C_1, \dots, C_n\}$ results in an argument in the VAF $\mathcal{F}$,*
- *attacks is an irreflexive and asymmetric binary relation on $\mathcal{C}$ which corresponds to the set of disagreement edges between users' opinions in $\mathcal{E}$:*

$$attacks = \{(C_a, C_b) \in \mathcal{E} \mid \mathcal{L}(C_a, C_b) < 0\};$$

  i.e. the user u_a attacks the user u_b if and only if $C_a \rightsquigarrow C_b$,
- *R is a non-empty set of values that models the social support of users,*
- *$Val : \mathcal{C} \to R$ is a valuation function that assigns social support values in R to arguments or users' opinion in $\mathcal{C}$, and*
- *$Valpref \subseteq R \times R$ is an order relation (transitive, irreflexive and asymmetric) on R reflecting preferences between social support values in R.*

Once we have the VAF $\mathcal{F}$ associated with a UDebG $\mathcal{G}$, we consider ideal semantics, formalized by Dung et al. [8], to define the set of widely accepted users' opinions from the debate, also called *the solution* for the debate. Ideal semantics guarantees that users' opinions in the solution represent the maximal set of acceptable user's opinions, in the sense that contrary opinions are weaker or they are defeated by other accepted stronger user's opinions, and are consistent, in the sense that there are no defeats between user opinion's in the solution, as we formally describe next.

In our approach, a defeat, or effective attack between two users' opinions, is defined relative to an attack strength threshold $\alpha \in [0, 2]$, which characterizes the minimum degree of disagreement of one user's opinion regarding another user's opinion, and relative to the strength (social support) of the user's opinions.

Formally, let $\mathcal{F} = \langle \mathcal{C}, attacks, R, Val, Valpref \rangle$ be a VAF for a UDebG $\mathcal{G} = \langle \mathcal{C}, \mathcal{E}, \mathcal{L} \rangle$ and let $\alpha \in [0, 2]$ be an attack strength threshold. The *defeats* relation over users based on the *Val* function, the *Valpref* relation and the threshold α is defined as follows:

$$defeats = \{(C_a, C_b) \in attacks \mid (Val(C_b), Val(C_a)) \notin Valpref \text{ and } |\mathcal{L}(C_a, C_b)| > \alpha\};$$

i.e. the user u_a *defeats* the user u_b if and only if (i) $C_a \rightsquigarrow C_b$, (ii) the social support value of C_b is not preferred over the social support value of C_a and (iii) the highest degree of disagreement of comments in C_a with respect to comments in C_b is greater than α.

Notice that the attack strength threshold $\alpha = 0$ has no pruning effect on the *defeats* relation.

Based on the *defeats* relation, we say that a set of users' opinion $S \subseteq \mathcal{C}$ is *conflict-free* if for all $C_a, C_b \in S$, $(C_a, C_b) \notin$ *defeats*. Moreover, a conflict-free set of users' opinion $S \subseteq \mathcal{C}$ is *maximally admissible* if for all $C_a \notin S$, $S \cup \{C_a\}$ is not conflict-free and, for all $C_b \in S$, if $(C_a, C_b) \in$ *defeats*, then there exists $C_d \in S$ such that $(C_d, C_a) \in$ *defeats*. Finally, *the solution* for a debate is the largest admissible conflict-free set of users' opinion $S \subseteq \mathcal{C}$ in the intersection of all maximally admissible conflict-free sets.

We select this semantics to define the solution for a debate, because it represents a maximally admissible set of conflict-free users' opinions, such that they defend against attacks outside this set, and they are included in any admissible set of users' opinions. This set therefore represents a kind of *maximum set of widely accepted user's opinions* between all the possible admissible sets of users' opinions. Dung et al. [8] prove that this solution is unique.

4.2. Implementation and analysis of results

As for the implementation purposes of our argumentation-based reasoning system, we instantiate the elements of a VAF $\mathcal{F} = \langle \mathcal{C}, \textit{attacks}, R, \textit{Val}, \textit{Valpref} \rangle$ as follows: (i) the set R of social support values of users is instantiated to the set of natural numbers $\mathbb{N}$, (ii) the preference relation *Valpref* on R is instantiated to the natural order on $\mathbb{N}$, and (iii) the valuation function *Val* maps a user's opinion $C_a \in \mathcal{C}$ to a natural number in $\mathbb{N}$. With the aim of stratifying users with really significant levels of relevance in the debate platform, the *Val* function stratifies the social relevance of users' opinions by assigning zero to a user's opinion with zero or negative support and, for positive support, considers that one user's opinion has more social relevance than another only if the support is at least ten times bigger. So, we consider the following definition for the valuation function *Val*: $\textit{Val}(C_a) = \lfloor \log_{10} \mathtt{support(C_a)} \rfloor + 1$, if $\mathtt{support(C_a)} \geq 1$ and $\textit{Val}(C_a) = 0$, otherwise, where $\mathtt{support(C_a)}$ is some social support metric for users' opinions available on the Reddit platform.

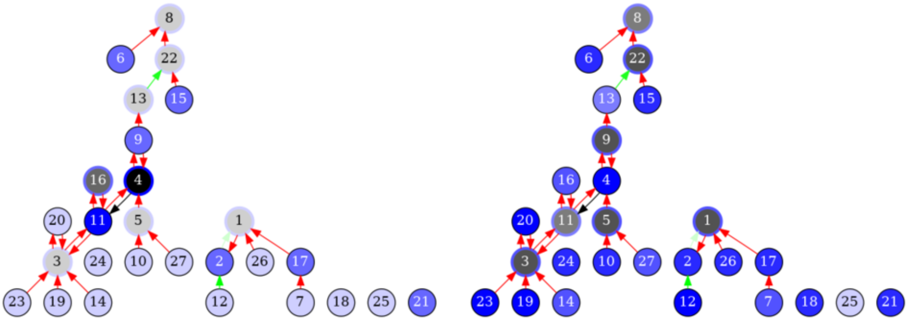
To compute the social support metric for users' opinions, we consider two different user parameters available on the Reddit platform. On the one hand, we consider a metric based on the *Comment Karma* global user parameter, refereed as $\mathtt{karma}(C_a)$ and computed as the Comment Karma of the user u_a available in Reddit. In this case, active users that post interesting comments are considered more relevant than users who are not very active or have little impact on the platform. On the other hand, we also consider a social support metric based on the *sum of the scores* of all the comments of a user in the debate under analysis and computed as $\mathtt{score}(C_a) = \sum_{\{(m, u_a, sc) \in C_a\}} sc$. In this case, users whose comments are valued or rated positively are considered more relevant than users with comments with negative scores.

Although the computational complexity of computing ideal semantics for general VAFs is even higher than NP, there are some cases that can be solved within polynomial time. With the goal of being able to solve instances with a scalable approach, we have developed a distributed algorithm [1] that computes within linear time the solution of a VAF for two special cases: graphs with no even cycles and bipartite graphs.

The implementation of the distributed algorithm consists of a pre-processing step and the actual distributed computation of the solution of the VAF resulting of the pre-processing. The pre-processing step prunes all attacks that are not effective based on the weighting scheme of edges and the valuation function of nodes. The distributed computation algorithm is based on the computation model of Pregel. This model is appropriate for our problem, because the input for a Pregel algorithm is a directed graph, where the nodes can be in different states, and the goal of a distributed algorithm in Pregel is to compute the state of each node based on the state of the nodes' neighbors.

Also, we have already shown [1] that for a VAF with a general graph, the output provided by the distributed algorithm coincides with the so-called *grounded* extension or solutionwhich, in turn, is always a subset of the ideal semantics solution (the solution we want to find). So, we can safely use the distributed algorithm as a sound *approximation* algorithm for such cases that are in principle not solvable by the algorithm (i.e. graphs with even cycles that are not bipartite).

Figure 1 shows the solutions for a Reddit debate of the subreddit World News (r/worldnews) [2] based on the `score` (left) and `karma` metrics (right). Each user's opinion is represented as a node and each relationship between users' opinions is represented as an edge. The edges are colored in black, green and red to denote neutral, agreement and disagreement relationships between users' opinions, respectively. The nodes colored in blue are the users' opinions in the solution and the nodes colored in gray are the rejected ones, where the darkness of the color is directly proportional to the value of the `score` and `karma` metrics of each user with respect to the maximum value.



(a) Solution based on the `score` metric.

(b) Solution based on the `karma` metric.

Figure 1. Solutions for a Reddit debate with attack strength threshold $\alpha = 0.5$.

For the `score` metric, the *solution* contains 18 of the 27 users' opinions. We observe that from the users that concentrate more interactions (3, 4, and 11), the solution includes only user 11, because user 11 has a score bigger than the ones of the users 3 and 16, so although there are mutual attacks between 11 and 3 and 11 and 16, only the ones from user 11 are defeats. For the `karma` metric, the solution changes because the relevance of some users changes. This time, we have that user 16, that attacks user 11, is more relevant, so user 16 defeats user 11, that in turn causes that user 4 is accepted in the solution. This impacts also on other users. For example, user 9 is not accepted because the accepted user 4 defeats user 9, and this in turn produces the acceptance of user 13. So, changes in the way of measuring the relevance of user's opinions may have a relevant contribution on the set of accepted opinions.

[2] `https://www.reddit.com/r/worldnews/comments/f62i35`

As future work, we plan to study the suitability of other schemes for the aggregation of individual interactions between two users, when computing the sentiment weighting scheme, as well as different approaches for the definition of the users' relevance.

Acknowledgments This work was partially funded by Spanish Project PID2019-111544GB-C22, by the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement 723596 and Grant Agreement 768824, and by 2017 SGR 1537.

References

[1] Alsinet, T., Argelich, J., Béjar, R., Cemeli, J.: A distributed argumentation algorithm for mining consistent opinions in weighted twitter discussions. Soft Comput. 23(7), 2147–2166 (2019)

[2] Alsinet, T., Argelich, J., Béjar, R., Esteva, F., Godo, L.: A probabilistic author-centered model for twitter discussions. In: Proceedings of IPMU Conference. vol. 854, pp. 683–695. Springer (2018)

[3] Alsinet, T., Argelich, J., Béjar, R., Martínez, S.: Measuring user relevance in online debates through an argumentative model. Pattern Recognition Letters 133, 41–47 (2020)

[4] Bench-Capon, T.J.M.: Value-based argumentation frameworks. In: Proceedings of NMR International Workshop. pp. 443–454 (2002)

[5] Burton, S., Soboleva, A.: Interactive or reactive? marketing with Twitter. Journal of Consumer Marketing 28(7), 491–499 (2011)

[6] Choi, D., Han, J., Chung, T., Ahn, Y.Y., Chun, B.G., Kwon, T.T.: Characterizing conversation patterns in reddit: From the perspectives of content properties and user participation behaviors. In: Proceedings of COSN Conference. pp. 233–243. ACM (2015)

[7] Chu, S.C., Kim, Y.: Determinants of consumer engagement in electronic word-of-mouth (ewom) in social networking sites. International Journal of Advertising 30(1), 47–75 (2011)

[8] Dung, P.M., Mancarella, P., Toni, F.: Computing ideal sceptical argumentation. Artif. Intell. 171(10-15), 642–674 (2007)

[9] Jansen, B.J., Zhang, M., Sobel, K., Chowdury, A.: Twitter power: Tweets as electronic word of mouth. Journal of the American Society for Information Science and Technology 60(11), 2169–2188 (2009)

[10] Manning,C.D.,Surdeanu,M.,Bauer,J.,Finkel,J.,Bethard,S.J.,McClosky,D.:TheStanford CoreNLP natural language processing toolkit. In: Association for Computational Linguistics (ACL) System Demonstrations. pp. 55–60 (2014)

[11] Martinelli, F., Santini, F.: Debating cybersecurity or securing a debate? In: International Symposium on Foundations and Practice of Security. pp. 239–246. Springer (2014)

[12] Martinelli, F., Santini, F., Yautsiukhin, A.: Network security supported by arguments. In: 2015 13th Annual Conference on Privacy, Security and Trust (PST). pp. 165–172. IEEE (2015)

[13] McCarthy, L., Stock, D.: How travelers use online and social media channels to make hotel-choice decisions. Cornell Hospitality Reports 10(18), pp. 6–18 (2010)

[14] Motamedi, R., Rezayi, S., Rejaie, R., Willinger, W.: On characterizing the twitter elite network. In: Proceedings of ASONAM Conference. pp. 234–241. IEEE (2018)

[15] ten Thij, M., Ouboter, T., Worm, D., Litvak, N., van den Berg, H., Bhulai, S.: Modelling of trends in twitter using retweet graph dynamics. In: Algorithms and Models for the Web Graph. pp. 132–147. Springer (2014)

[16] Villatoro, D., Serna, J., Rodríguez, V., Torrent-Moreno, M.: The tweetbeat of the city: Microblogging used for discovering behavioural patterns during the mwc2012. In: Citizen in Sensor Networks, LNCS, vol. 7685, pp. 43–56. Springer (2013)

[17] Williams, N.L., Inversini, A., Buhalis, D., Ferdinand, N.: Community crosstalk: an exploratory analysis of destination and festival ewom on Twitter. Journal of Marketing Management 31(9-10), 1113–1140 (2015)

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210124

Argumentation Reasoning with Graph Neural Networks for Reddit Conversation Analysis

Teresa ALSINET, Josep ARGELICH, Ramón BÉJAR [1], Daniel GIBERT,
Jordi PLANES and Nil TORRENT
INSPIRES Research Center – University of Lleida
Jaume II, 69 – 25001 Lleida, SPAIN

Abstract The automated analysis of different trends in online debating forums is an interesting tool for sampling the agreement between citizens in different topics. In these online debating forums, users post different comments and answers to previous comments of other users. In previous work, we have defined computational models to measure different values in these online debating forums. A main ingredient in these models has been the identification of the set of *winning posts* trough an argumentation problem that characterizes this winning set trough a particular argumentation acceptance semantics. In the argumentation problem we first associate the online debate to analyze as a debate tree. Then, comments are divided in two groups, the ones that agree with the root comment of the debate, and the ones that disagree with it, and we extract a bipartite graph where the unique edges are the disagree edges between comments of the two different groups. Once we compute the set of winning posts, we compute the different measures we are interested to get from the debate, as functions defined over the bipartite graph and the set of winning posts. In this work, we propose to explore the use of graph neural networks to solve the problem of computing these measures, using as input the debate tree, instead of our previous argumentation reasoning system that works with the bipartite graph. We focus on the particular online debate forum Reddit, and on the computation of a measure of the polarization in the debate. Our results over a set of Reddit debates, show that graph neural networks can be used with them to compute the polarization measure with an acceptable error, even if the number of layers of the network is bounded by a constant.

Keywords. Reddit, social networks analysis, argumentation, graph neural networks

1. Introduction

Recently, there has been a growing interest in the use of Graph Neural Network (GNN) approaches to model and solve reasoning problems defined via graph inputs [10,12,17]. The most common approach used by a GNN is to map the feature vector of each node

[1] Correspondence to: R. Béjar. INSPIRES Research Center, University of Lleida. C/Jaume II, 69. Lleida, Spain. Tel.: +34 973703477; E-mail: ramon.bejar@udl.cat.

to an embedding representation that also uses (by aggregation) the feature vector of its neighbor nodes. By iterating this scheme k times, the final representation of each node tends to capture structural information within the node's $k-$hop neighborhood. This scheme can be used to learn any kind of function over graphs that outputs a labeling of its nodes, or that outputs a single value (for graph classification tasks).

In previous work, we have considered the use of argumentation based models to analyze different characteristics of social network debates. In the argumentation based approach, we first identify a valued argumentation problem with the debate to be solved, where debate posts are associated with arguments, under a particular acceptance semantics: a set of rules that define what arguments are accepted and what are rejected. The usual acceptance semantics tend to be NP-hard, like the *ideal semantics* [8] we have used in our previous works about measuring discussion polarization with argumentation based models [2,3].

In this work we initiate a line of investigation to study whether a GNN approach can be a good candidate to solve argumentation-based problems with less effort. Our focus is not on exactly replicating the set of accepted arguments of the discussion, as it has been already explored on recent work about solving some abstract argumentation problems with GNNs [6,13], but on being able to compute the final measure of interest defined from the set of accepted arguments. Our hypothesis is that even if the worst-case complexity of computing accepted arguments is in general NP-hard, it may be possible to compute, or approximate, the final measure with much less computational effort. In particular, in this work we focus on the computation of a measure of discussion polarization that is defined in function of the set of accepted arguments of a discussion, and whether these arguments agree or disagree with the root topic of the discussion. Our discussions come from the social network Reddit. A Reddit debate is first represented as a debate tree, where edges represent agreement or disagreement relationships between Reddit posts. Then, this debate tree is processed to get a bipartite debate graph where posts are divided in two groups: the ones that agree with the root comment of the debate, and the ones that disagree with it. The edges of the bipartite graph represent disagreement between comments of the two groups.

Our results show that we can devise a reasoning system to compute that polarization measure, defined initially from the set of accepted arguments and the bipartite debate graph, based only on the original debate tree (the graph previous to the bipartite graph) and that computes the polarization measure with acceptable error, without explicitly computing the set of accepted arguments of the associated argumentation problem.

The structure of the paper is as follows. In Section 2 we present the relevant definitions for our argumentation-based Reddit analysis system. In Section 3 we briefly survey previous results about GNNs. In Section 4 we present the GNN architecture we have used to model our reasoning system. Finally, in Section 5 we present the experimental results we have obtained with a dataset of Reddit debates.

2. Reddit Debate Analysis

In this section we give the definitions of the different components of the Reddit analysis system introduced in [3]. It is based on two main components: a Reddit debate retrieval system and an argumentation-based reasoning system. The retrieval system takes a root comment and obtains the complete set of comments generated in the debate on that root comment.

Definition 1 *A* comment *c is a tuple $c = (m, u, sc)$, where m is the text of the comment, u is the user's identifier of the comment, and $sc \in \mathbb{Z}$ is the score of the comment.*

Let $c_1 = (m_1, u_1, sc_1)$ and $c_2 = (m_2, u_2, sc_2)$ be two comments. We say that c_1 answers *c_2 if c_1 is a reply to comment c_2.*

Let $r = (m_r, u_r, sc_r)$ be a comment such that m_r contains a link to some news. A Reddit debate *on the (root) comment r is a non-empty set Γ of Reddit comments such that $r \in \Gamma$ and every comment $c \in \Gamma$, $c \neq r$, c answers some comment in Γ*[1].

Next, we obtain the tree representation of a Reddit debate where we incorporate edge labels that express the sentiment of the comments.

Definition 2 *Let Γ be a* Reddit debate *on a (root) comment r. The* Debate Tree (DebT) *for Γ is a tuple $\mathcal{T} = \langle C, r, E, L \rangle$ such that:*

- *for every comment in Γ there is a node in C,*
- *node $r \in C$ is the root node of $\mathcal{T}$,*
- *if c_1 answers c_2 then there is a directed edge (c_1, c_2) in E, and*
- *L is a labeling function $L : E \rightarrow [-2, 2]$, where the value assigned to an edge denotes the sentiment of the answer, from highly negative (-2) to highly positive (2).*

Only the nodes and edges obtained by applying this process belong to C and E, respectively.

As argued in [3], we consider in our model that subtrees with a neutral root do not contribute anything relevant with respect defending or rejecting the root comment of the debate. So, the next step is to prune out those subtrees with respect to a pruning threshold.

Definition 3 *Let α be a pruning threshold in the real interval $[0, 2]$ and let $\mathcal{T} = \langle C, r, E, L \rangle$ be a* DebT. *The* Pruned Debate Tree (PDebT) *for $\mathcal{T}$ with respect to α is a tuple $\mathcal{T}_\alpha = \langle C_\alpha, r, E_\alpha, L \rangle$, where both sets of pruned comments $C_\alpha \subseteq C$ and pruned edges $E_\alpha \subseteq E$ are defined as follows:*

- *the root node (comment) $r \in C_\alpha$,*
- *r is the root node of $\mathcal{T}_\alpha$ and*
- *if $(c_1, c_2) \in E$ with $c_2 \in C_\alpha$, then $c_1 \in C_\alpha$ and $(c_1, c_2) \in E_\alpha$, whenever $|L(c_1, c_2)| \geq \alpha$.*

Only the nodes and edges obtained by applying this process belong to C_α and E_α, respectively.

Note that for $\alpha = 0$ the pruning threshold has no effect, in the sense that the PDebT obtained corresponds to the original DebT and that for $\alpha = 2$ the PDebT obtained only contains strictly polarized both positive and negative answers. In any case, the PDebT $\mathcal{T}_\alpha$ is a subtree of $\mathcal{T}$ with r being the root node.

[1] Given the structure of a Reddit debate, except for the root comment, each comment answers exactly one previous comment, usually by another user or author.

Finally, we divide the set of comments into two sets: comments supporting the root comment and comments that disagree with it. Then, the attacks between the comments of both sets are defined as a subset of edges in E_α such that they are negative answers from a comment in one of the sets to a comment in the other set, obtaining a bipartite graph that represents both sides of the debate, and the disagreement between them. This bipartition can be computed with the algorithm that we presented in [2]. Moreover, we also label each node of the graph obtained with a weight that denotes the comments' social acceptance during the debate. Next we formalize the Weighted Bipartite Debate Graph structure.

Definition 4 *Let* $\mathcal{T}_\alpha = \langle C_\alpha, r, E_\alpha, L\rangle$ *be a* PDebT *for a Reddit debate* Γ. *A* Weighted Bipartite Debate Graph (WBDebG) *for* $\mathcal{T}_\alpha$ *is a tuple* $G = \langle C_+ \cup C_-, E_-, W\rangle$ *where*

- C_+ *and* C_- *is a bipartition of* C_α. *Thus,* $C_+ \cup C_- = C_\alpha$ *and* $C_+ \cap C_- = \emptyset$, *where* C_+ *denotes the set of comments that agree with the root comment* c_r, *and* C_- *denotes the set of comments that disagree with it.*
- $E_- = \{(c_1, c_2) \in E_\alpha \mid L(c_1, c_2) < 0\}$ *and corresponds with the set of disagreement edges between the comments in* C_+ *and* C_-. *Thus, if* $(c_1, c_2) \in E_-$, *then either* $c_1 \in C_+$ *and* $c_2 \in C_-$ *or,* $c_1 \in C_-$ *and* $c_2 \in C_+$.
- W *is a weighting scheme* $W : C_\alpha \to \mathbb{N}$ *of the weight of nodes (comments). The weighting scheme* W *evaluates the social acceptance of comments by mapping the score sc of a comment* $(m, u, sc) \in C_\alpha$ *to a value in* $\mathbb{N}$.

At this point we are ready to introduce the argumentation-based reasoning system used to obtain the set of comments, from the two opposite groups of a WBDebG, that are accepted in the sense that this set should represent a kind of consensus among all the comments of the debate. To this end, we use value-based abstract argumentation [5] to model the weighted argumentation problem associated with a WBDebG and ideal semantics [7] to compute its solution (the set of comments that can be accepted).

The *value-based abstract argumentation framework* (VAF) we define for a WBDebG $G = \langle C_+ \cup C_-, E_-, W\rangle$, interprets each comment in $C_+ \cup C_-$ as an argument and defines a *defeat* relation (or effective attack relation) between arguments as follows:

$$\textit{defeats} = \{(c_1, c_2) \in E_- \mid W(c_{2)}) \not> W(c_1)\};$$

i.e. argument c_1 *defeats* argument c_2 if and only if c_1 attacks or disagrees with c_2 and the social acceptance value of c_2 is not preferred over the social acceptance value of c_1, based on the weighting scheme W.

Then, a set of comments $S \subseteq C_+ \cup C_-$ is called *conflict-free* if for all $c_1, c_2 \in S, (c_1, c_2) \notin$ *defeats*, and a conflict-free set of comments $S \subseteq C_+ \cup C_-$ is defined as *maximally admissible* if for all $c_1 \notin S$, $S \cup \{c_1\}$ is not conflict-free and, for all $c_2 \in S$, if $(c_1, c_2) \in$ *defeats* then there exists $c_3 \in S$ such that $(c_3, c_1) \in$ *defeats*. Finally, the *solution* or *set of accepted comments* for a debate is the largest admissible conflict-free set of comments $S \subseteq C_+ \cup C_-$ in the intersection of all maximally admissible conflict-free sets.

We select this semantics to define the solution for a debate, because it represents a maximally admissible set of conflict-free comments, such that they defend against attacks outside the set with comments inside the set, and they are included in any admissible set of comments. This set therefore represents a kind of *maximum consensus* between all the possible admissible sets of comments. For our particular case of an acyclic VAF, the picture is even simpler, as there is a unique maximally admissible set, and thus the solution for ideal semantics coincides with this set. Moreover, for the case of a VAF that is acyclic or bipartite (as in the case of a WBDebG), we can compute its solution in linear time, with respect to the number of comments, for instances of big size with the distributed algorithm we developed in [1]. However, in the worst case the status of each comment in the solution may depend on the status of the rest of the comments, so that is why we explore in this work a possible GNN-based architecture where nodes (comments) only consider information from nodes at distance bounded by a constant.

Given that the solution for the debate provides us with a consensus point of view, an interesting characteristic to analyze is its degree of polarization.

Definition 5 *Let* $G = \langle C_+ \cup C_-, E_-, W\rangle$ *be a* WBDebG *and let* $S \subseteq C_+ \cup C_-$ *be the solution for* G*. The* polarization degree *of solution* S *is a measure in the real interval* $[-1, 1]$ *defined as follows:*

$$\textit{polarization}(S) = \frac{\#(S \cap C_+) - \#(S \cap C_-)}{\#S}.$$

We use the polarization degree value as a measure of the bias of the solution S towards comments in C_+ and comments in C_-. The value that indicates total bipartisanship (0) is obtained when the number of comments of S in C_+ equals the number in C_-. The highest positive value is obtained when all the comments of the solution are found in C_+, and analogously for the lowest negative value. In order to classify debates in terms of the polarization degree, instead of this measure, we can also work with a more qualitative measure mapping from it, to a discrete set of values. For example, in this work we stratify Reddit debates in five levels, based on the polarization degree of the solution:

$$\textit{bias-level} : \textit{polarization}(S) \rightarrow \{-2, -1, 0, 1, 2\}.$$

3. Graph Neural Networks

In the last years, there has been an increasing interest in analyzing graphs with machine learning (ML) [9,16] because of the immense expressive power of graphs, i.e. graphs can be used to model the interaction between complex structures such as proteins, mRNA, particles in physics models, etcetera. Thus, a key factor to be considered when dealing with graphs using ML is the ability of the methods to deal with graphs of different sizes and shapes.

There have been various attempts in the literature using graph neural networks (GNNs), mainly by: (i) focusing on learning node embeddings by aggregating the nodes, and (ii) by mapping from the node neighbourhood domain (adjancency matrix) to spectral domain. From the first type, we highlight the Generalizing Aggregation GraphSage [10] used for node classification. This method focuses on learning node embeddings, and then a model aggregates the resulting embeddings to handle size-varying neighbourhoods. From the second type, we feature the Spectral Graph Convolution Model [12] used for the classification of nodes using their adjacency matrix. In addition, it uses Chebyshev filters (passband filters) and Lapacian regularization in the loss function.

A recent improvement over the first method are Graph Isomorphism Networks (GIN), presented in a study [17] of GNN expressivity w.r.t. Wesfeiler-Lehman (WL) test [15] of graph isomorphism, where they proposed a WL equivalent aggregator, i.e. it generalizes the WL test and thus, it achieves the maximum discriminative power among the GNNs in the literature.

4. GNN Modelling

We propose the use of Graph Isomorphism Networks (GIN) [17] in our GNN model to approximately compute the polarization degree of a Reddit debate. In particular, our GIN model receives as input a Pruned Debate Tree (PDebT) $\mathcal{T}_\alpha = \langle C_\alpha, r, E_\alpha, L \rangle$ with $|C_\alpha| = N$ nodes, obtained from a Reddit debate as explained in [2], and outputs a bias-level of the polarization degree.

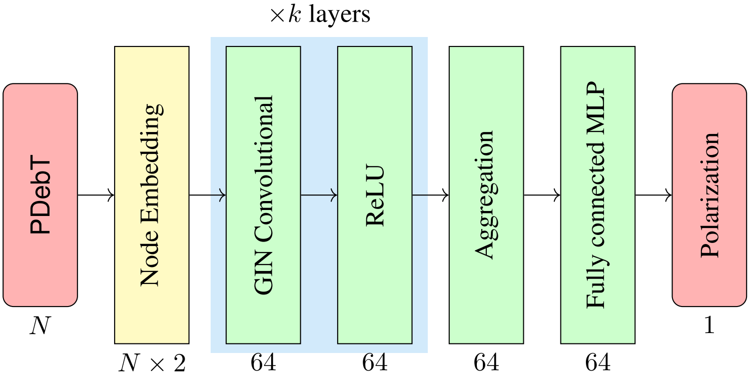


Figure 1. GNN architecture for computing the polarity degree of a Pruned Debate Tree.

The overall architecture is presented in Figure 1. It comprises the following layers:

Node embedding The input layer contains a two dimensional vector for each non-root comment $c_i = (m_i, u_i, sc_i)$ that contains the score of the comment sc_i and the sentiment from the label $L(c_i, c_j)$, where c_j is the unique comment such that $(c_i, c_j) \in E_\alpha$.

GIN Convolutional (k layers). Every layer combines the node embedding of the previous layer considering the node close neighbours. The aggregator in the layer l is the following:

$$\mathbf{x_i}^{(l)} = MLP\left((1+\epsilon)\cdot \mathbf{x_i}^{(l-1)} + \sum_{j \in \mathcal{N}(i)} \mathbf{x_j}^{(l-1)}\right)$$

where $\mathbf{x_i}^{(l)}$ is the embedding of node i in layer l, ϵ is a learnable parameter, and MLP is a multi-layer perceptron with nonlinearity, and $\mathcal{N}(i)$ is the set of neighbours of node i. The first GIN layer has an input dimension of 2 and an output dimension of 64. The following layers have input and output dimensions of 64. Globally, this GIN block maps the two-dimensional vector of each node to a vector of 64 values that tries to capture the information from nodes k hops away from it. Also, we insert a Rectified Linear Unit (ReLU) layer after each GIN layer, to help encode non-linear outputs in the network.

Normalization We give also the option to include a normalization layer between consecutive GIN layers, because previous work suggests that it may speed up the learning process [4].

Aggregation The aggregation layer creates the final graph representation using the mean operator, aggregating all the node embeddings into one graph embedding, as a vector with the same dimension (64).

Fully connected MLP This block maps the final aggregated embedding representation of the graph into the polarization bias-level of the debate.

After every ReLU layer and at the end of the fully connected MLP, a dropout of 0.25 is applied to prevent overfitting [11]. We use the pytorch and pytorch geometric python libraries to implement this GNN model.

5. Experimental results

In this section we present the results obtained when learning a GNN model with the GIN architecture introduced in Section 4 to compute the polarization bias-level for a set of Reddit debates.

To train and test our models, we use a dataset with 40 Reddit debates, where 34 have been used for training and 6 have been used for testing. To download the set of comments for each Reddit debate we use the Python Reddit API Wrapper (PRAW) [2]. Then, in the PDebT $\mathcal{T}_\alpha$ obtained from each Reddit debate, the label for each edge (c_1, c_2) is computed with the sentiment analysis software of [14]. It uses the text of the comment c_1, where the value assigned denotes the sentiment of the answer, from highly negative (-2) to highly positive (2). The pruning parameter α is set to the value 0.15. We have tried three different values for the number of GIN layers $(2, 4, 6)$ and also experimented with either using a normalization layer after each GIN layer or not. The number of GIN layers is kept low, compared with the number of nodes of the graphs that ranges from 5 to 4472 nodes, to explore whether bounding the neighborhood size used by the GNN still allows a reasonable approximation of the right output value. As we prefer a GNN model where the output value is as closer as possible to the right polarization bias-level, we train our GNN models using as the loss function the mean square error.

The experimental results for the average loss for the training set and the average loss for the test set are shown in Table 1, where each experiment was repeated with two different number of epochs $(250, 500)$ and executed 10 times (generating each time a different training/test set). The results shown in the table for each experimental setting are the best ones (with respect to the test set loss) from the set of 10 executions. The results show that the training loss is slightly higher than in the test set, suggesting that our models seem to not overfit with the training set. The results obtained with different

[2] https://github.com/praw-dev/praw

number of GIN layers do not set seem to have a significant impact on the test set loss when considering 500 epochs for learning. Analogously, the use of normalization layers between GIN layers do not seem to have a significant impact, as with no normalization the results are slightly better.

		Training Loss		Test Loss	
Num GIN layers	Normalization	Epochs 250	Epochs 500	Epochs 250	Epochs 500
2	True	0.32	0.35	0.37	0.18
4	True	0.42	0.33	0.15	0.28
6	True	0.47	0.33	0.24	0.23
2	False	0.45	0.47	0.27	0.04
4	False	0.50	0.27	0.30	0.11
6	False	0.34	0.44	0.18	0.12

Table 1. Experimental results for polarization computation with our GNN model.

To check whether our GNN model generalizes well when the number of nodes of the input increases, we have repeated the previous experiment, but only with no normalization and number of epochs 500, fixing as the training set the graphs with the smallest size (from 5 to 414 nodes in our case) and as the test set the biggest ones (from 1029 to 4472 nodes). The results obtained show that we have a slight increase in the test set average loss: 0.26 for 2 GIN layers, 0.27 for 4 and 0.23 for 6. So, at least with this test set, training with the smallest ones does not seem to increase significantly the test set loss, although these results should be further confirmed with larger training and test sets.

6. Conclusions

In this paper, we have presented a GNN-based system to solve the problem of computing a polarization measure from a Reddit debate. Our GNN-based system is based on our previous work, where we used an argumentation approach to solve this problem. Although we do not use the GNN architecture to explicitly compute the solution of the argumentation problem, it is able to approximate the final polarization measure, that it is originally defined from that solution. This happens even if our GNN model aggregates information in each node considering always a neighborhood with bounded distance, given that the number of GIN layers is kept constant.

An interesting direction for future work is to consider the computation of other argumentation-based measures that consider as input author graphs, instead of debate trees. Author graphs come from the aggregation of comments from the same author in a single node, such that the resulting graph may contain cycles, and in that case the complexity of the argumentation-based reasoning algorithm is higher than the one for the acyclic graphs we have considered in this work. Also, we plan to work with a bigger Reddit dataset to get more significant results.

Acknowledgments This work was partially funded by Spanish Project PID2019-111544GB-C22, by the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement 723596 and Grant Agreement 768824, and by 2017 SGR 1537.

References

[1] T. Alsinet, J. Argelich, R. Béjar, and J. Cemeli. A distributed argumentation algorithm for mining consistent opinions in weighted twitter discussions. *Soft Comput.*, 23(7):2147–2166, 2019.

[2] T. Alsinet, J. Argelich, R. Béjar, and S. Martínez. An argumentation approach for agreement analysis in reddit debates. In *Artificial Intelligence Research and Development - Current Challenges, New Trends and Applications, CCIA 2018, 21st International Conference of the Catalan Association for Artificial Intelligence, Alt Empordà, Catalonia, Spain, 8-10th October 2018*, pages 217–226, 2018.

[3] T. Alsinet, J. Argelich, R. Béjar, and S. Martínez. Measuring user relevance in online debates through an argumentative model. *Pattern Recognit. Lett.*, 133:41–47, 2020.

[4] L. J. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *CoRR*, abs/1607.06450, 2016.

[5] T. J. M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.

[6] D. Craandijk and F. Bex. Deep learning for abstract argumentation semantics. In C. Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 1667–1673. International Joint Conferences on Artificial Intelligence Organization, 7 2020. Main track.

[7] P. M. Dung, P. Mancarella, and F. Toni. Computing ideal sceptical argumentation. *Artif. Intell.*, 171(10-15):642–674, 2007.

[8] P. E. Dunne. The computational complexity of ideal semantics. *Artif. Intell.*, 173(18):1559–1591, 2009.

[9] F. Errica, M. Podda, D. Bacciu, and A. Micheli. A fair comparison of graph neural networks for graph classification. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[10] W. L. Hamilton, Z. Ying, and J. Leskovec. Inductive representation learning on large graphs. In I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 1024–1034, 2017.

[11] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *CoRR*, abs/1207.0580, 2012.

[12] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[13] I. Kuhlmann and M. Thimm. Using graph convolutional networks for approximate reasoning with abstract argumentation frameworks: A feasibility study. In N. Ben Amor, B. Quost, and M. Theobald, editors, *Scalable Uncertainty Management*, pages 24–37. Springer International Publishing, 2019.

[14] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60, 2014.

[15] B. Weisfeiler and A. Leman. A reduction of a graph to a canonical form and an algebra arising during this reduction. *Nauchno-Technicheskaya Informatisiz*, 2(9):12–16, 1968.

[16] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu. A comprehensive survey on graph neural networks. *IEEE Trans. Neural Networks Learn. Syst.*, 32(1):4–24, 2021.

[17] K. Xu, W. Hu, J. Leskovec, and S. Jegelka. How powerful are graph neural networks? In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210125

Learning Cognitive-Affective Digital Twins from Social Networks

Andrea NAVARRO [a], Rafael BERLANGA [a] and Lledó MUSEROS [a]
[a] *Universitat Jaume I, Spain*

Abstract. This paper presents an ongoing project about the implementation of digital twins (DT) for simulating cognitive-affective behaviours in social networks. Our approach relies on a pure data-driven solution, which takes existing public data from social networks to learn cognitive models according to the profile, posts and interactions of the social network users. The final aim is that the learned cognitive models can be parameterised according to existing classifications of traits and emotions so that different behaviours can be eventually simulated with the resulting DTs. In this work, we propose the use of the Transformers neural-network architectures to both interpret incoming messages according to cognitive contexts, and to generate responses to these messages. The first experiments are aimed at integrating and measuring existing approaches for emotion recognition from texts.

Keywords. Digital Twin, Social Networks, Emotion and Personality Recognition

1. Introduction

Digitization is a common and global trend that companies are investing in after a decade with higher computing capabilities, better communication infrastructures, more sensors and a large amount of available data. It is in this context where Digital Twins (DT) are starting to be deployed in companies. DT [1] can be defined as (physical and/or virtual) machines or computer-based models that are simulating, emulating, mirroring, or twinning the life of a physical entity, which may be an object, a process, a human, or a human-related feature. When the system to be replicated is very complex, learning a model from the data generated by the system is the preferred approach to emulate its behavior. At the moment, DTs are widely used in manufacturing to optimize asset performance, improve process efficiency, and minimize time and costs. Extrapolating smart capabilities from a DT approach to other sectors is challenging, not only from an implementation perspective but also from an ethical and social point of view. Thus, a *Cognitive Digital Twin* can be defined as a digital representation, augmentation, and intelligent companion of its physical twin as a whole, including its subsystems and across all of its life cycles and evolution phases"[1]. [2] focuses on DT for reproducing human cognitive processes in cyber-simulation. They define Cognition DT as a model that monitors, and predicts a persons cognitive status through the processing of different type of information. Finally, on the context of social media, [3] the DT paradigm has been considered to establish a

[1] https://www.linkedin.com/pulse/emergence-cognitive-digital-physical-twins-cdpt-21st-ahmed/

link among social media data analysis for a virtual product. This research has attempted to use AI tools to categorize the sentiment trends and fill the gap for the relationship between user emotions and product design. Unfortunately, there are no approaches for designing DTs that takes into account both the cognitive and the affective perspectives. This paper presents a first approach towards the definition and implementation of such a cognitive DT, specially focused on exploiting the large amount of social network data by applying new deep learning methods to learn the main DT components.

2. Methodology

Figure 1 sketches our proposal for designing cognitive-affective DTs. This proposal basically follows the stimulus-organism-response (SOR) scheme, which has been successfully applied to a wide range of fields from psychology to marketing. In our scenario, the scheme is adapted to social network data, as these data will guide the learning process of the different models involved in the digital twin. Thus, the stimulus consists of some message or image whose contents are interpreted according to the *temper* of the twin. This interpretation results in a particular configuration in the *attitude* model, which favours the acceptance of the stimulus as well as the direction this decision will be projected towards. Acceptance-rejection dimension is usually included in most sentiment/emotion models (e.g., polarity). To accept an idea means that it is well aligned with our cognition space, whereas to reject an idea means that it conflicts somehow with our cognition space. The second dimension of the *attitude* model is the direction towards the decision is projected. If the direction is towards oneself, then the emotion will be projected on the oneself behaviour (e.g., feeling shame, pride, etc.) If the direction is towards others, then the emotion will be projected over an object, person, community, and so one (e.g., hate, gratitude, etc.). Once the stimulus is processed in the cognitive component, the *affective* component is activated. In the *affective* component, emotions are selected according to the result of the cognitive results. The model for the *affective* component has been simplified to five major emotions and a neutral class. Emotions are organised according to two excluding class pairs: ”joy vs. sadness” and ”fear vs. anger”. The ”disgust” class has no counterpart and is not dependent of the others. From the results of the cognitive and affective components, a final reaction is generated. The reaction is guided by the selected emotions and can consists of a message or an action over the input. For example, actions like retweeting, giving a like or replying a message are possible reactions to the input stimulus.

Let’s describe an example to illustrate this process. This starts with a stimulus like a dark humour post in social media. After the combination of this fact with the temper traits, the information will be processed and biased to obtain its attitude, which in this hypothetical case could be located on the low acceptance and outwards projection. That attitude would generate more likely emotions, suppose: sadness and annoyance (low intensity anger). Then, a message is released as reaction: I do not get how they can make fun of suffering, people are so insensitive.

2.1. Learning components

The *temper* component of the twin is modelled with the big five variables (i.e., openness, agreeableness, extroversion, conscientiousness, and neuroticism). Temper is a parameter

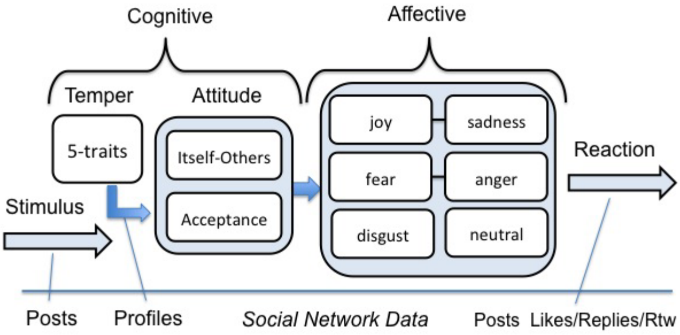


Figure 1. Structure of the cognitive-affective DT.

of the digital twin so that different personalities can be simulated with this component. The learning goal of this component is to associate the big five variables to the way personality transforms the attitude space. For example, a closed-hostile-conscientious personality will likely lead to negative perceptions and an inwards projection. Some work has been carried out for analysing the big five from users activity in social networks [4]. However, this type of experiments require to previously evaluate the personality of the users through proper questionnaires. Moreover, experiments are performed in controlled scenarios so collected data is small.In our proposal, we aim at learning models from existing data in social networks. In this case, the combination of data from users profiles and their messages can be a good indicator for finding some prototypical schemes for temper. Profiling is an active area in social network analysis and its techniques could be extended to this scenario [5].

Transforming the stimulus to the *attitude* space is a challenging problem. Given the representations of both the stimulus and the temper, we need to generate the values for the two attitude dimensions. Regarding the acceptance-rejection dimension, we can find lots of approaches that classify messages polarity, most of them applied to social network analysis. State-of-the-art approaches could be adapted in order to include temper as a condition. Regarding the attitude projection, as far as we know is has not been trhttps://www.overleaf.com/project/60a271d38920737e80cfdf69eated in the literature yet. Transforming *attitude* to emotions has been usually done with hand-written rules. For example, in the hourglass model [6] authors directly map emotions to the *attitude* space. However, some of these mappings are dependent to the temper component and the input stimulus, and cannot be mapped a priori. Thus, it is necessary to learn a model able to associate the cognitive and *affective* components for particular inputs and tempers.

Finally, the generation of the reaction will mainly rely in text generation. It is worth mentioning that mild positive emotions will usually lead to actions like retweet or give a like to the input message. More extreme emotions will likely lead to replies or new messages to express them.

2.2. Experiments design

Current state-of-the-art approaches in natural language processing relies on deep learning methods like recurrent neural networks and transformers architectures. Transform-

Dataset	Size	Emotions (missing)	Metrics	Competition/Year
GoEmotion	58K	21 (0)	BERT f1=0,46	Google Reseach 2020
Affect in Tweets	21K	11 (0)	RNN acc=0.588	SemEval 2018
Huggingface	20K	4 (disgust)	RoBERTa acc=0.93	(not published)

Table 1. Large datasets for emotion classification of texts.

ers architectures like BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained transformer) have become very popular thanks to their transfer-learning capabilities. This means that an initial model auto-trained with a very large dataset is then fine-tuned to perform different tasks like sentiment analysis, textual similarity and textual entailment. In the context of our proposal, we have explored existing work for detecting and classifying emotions from texts (see Table 1). Currently, we are collecting text samples from different domains (e.g., automotive, medicine, tourism and fashion) in order to train the cognitive-affective components not covered by these datasets.

3. Conclusions

In this position paper we describe how cognitive-affective DTs can be structured by means of a series of learning models, which interact to each other to react to stimuli. Preliminary experiments show that a few existing approaches work reasonably well for a very abstract classification of primal emotions, but fail when finer classifications are required. They also show that learning affective models needs to regard the cognitive context. This is because the interpretation of reactions and emotions are dependent on the application domain as well as on cultural aspects. Next steps are focused on building the datasets for training and validating the different components of the DT, mainly those not currently covered by the literature.

Acknowledgments: This project has been funded by the Ministry of Economy and Commerce with project contract TIN2016-88835-RET and by the Universitat Jaume I with project contract UJI-B2020-15.

References

[1] B. R. Barricelli, E. Casiraghi and D. Fogli: A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications. IEEE Access 7: 167653-167671. (2019)

[2] J. Lu, Z. Xiaochen, K. K. Ali Gharaei, D. Kiritsis: Cognitive Twins for Supporting Decision-Makings of Internet of Things Systems. ArXiv abs/1912.08547 (2019)

[3] A. A. Olad and O. F. Valilai: Using of Social Media Data Analytics for Applying Digital Twins in Product Development. 2020 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM) (2020): 319-323

[4] W. Youyou, M. Kosinski, D. Stillwell: Computer-based personality judgments are more accurate than those made by humans. PNAS 112(4): 1036-1040. (2015)

[5] I. Lanza-Cruz, R. Berlanga, M.J. Aramburu: Modeling Analytical Streams for Social Business Intelligence. Informatics 5: 33. (2018)

[6] Y. Susanto, A. G. Livingstone, B. C. Ng and E. Cambria: The Hourglass Model Revisited. IEEE Intelligent Systems 35(5): 96-102. (2020)

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210126

A Common Formal Framework for Factorial and Probabilistic Topic Modelling Techniques

Karina GIBERT [a,1], Yaroslav HERNANDEZ-POTIOMKIN [a]
[a] *Intelligent Data Science and Artificial Intelligence Research Group, Universitat Politècnica de Catalunya (Spain)*

Abstract. Topic modelling is nowadays one of the most popular techniques used to extract knowledge from texts. There are several families of methods related to this problem, among them 1) Factorial methods, 2) Probabilistic methods and 3) Natural Language Processing methods. In this paper a common conceptual framework is provided for Factorial and probabilistic methods by identifying common elements and describing them with common and homogeneous notation and 7 different methods are described accordingly. Under a common notation it is easy to make a comparative analysis and see how flexible or more or less realistic assumptions are made by the different methods. This is the first step to a wider analysis where all families can be related to this common conceptual framework and to go in depth in the understanding of stengths and weakenesses of each method and ellaboration of general guidelines to provide application criteria. The paper ends with a discussion comparing the presented methods and future research lines.

Keywords. Topic modelling, Multivariate methods, Probabilistic Methods

1. Introduction

Textual data analysis is a constantly growing field with many open research problems. Often, textual data analysis is used for 1) Understand the underlying topics of a set of documents and 2) Find the principal concepts that better summarize a given text.

There are many applications of topic modelling, such as enhanced document clustering [1], topic trend detection [14] high-dimensional classification [28] and dimensionality reduction and projection [18] among others.

However, currently there are some limitations of existing techniques, as for example considering semantic structures of a text; i.e. synonymy and polysemy, ortography, bias due to outliers and the fact that the topics are implicit and require subjective interpretation.

[1] Corresponding Author: Karina Gibert, Intelligent Data Science and Artificial Intelligence Research Group, Universitat Politècnica de Catalunya (Spain); E-mail: karina.gibert@upc.edu.

Topic modelling techniques mainly belong to three big families of methods, **1)** Factorial methods, **2)** Probabilistic methods and **3)** Natural Language Processing (NLP) methods. The former, aim to perform a decomposition over the multivariate design matrix in such a way that a given objective function is maximized and the given set of constraints are satisfied [6]. Several examples are Latent Semantic Analysis (LSA) [9] or Principal Components Analysis (PCA) [20], [30], Non-negative Matrix Factorization (NMF) [21], Canonical Correlation Analysis (CCA), Correspondence Analysis (CA) [20], Multiple Correspondence Analysis (MCA) [15], Non-linear Iterative Partial Least Squares algorithm (NIPALS) [30], Archetypal Analysis (AA) [6] among others.

Probabilistic methods, in turn, relies on statistical model definition, composed of probability model and a parameter's space definition. The parameters can be estimated either through the Maximum Likelihood function (*frequentist* approach), or *Bayesian* framework [12]. The probabilistic approaches are very valuable as they are generative, they provide clear interpretation, flexibility and extensibility. Research addressed to probabilistic topic modelling is widely applied to multi-document summarization [6], text classifiers [31], topics extraction [16] and topic-based document classification [23].

The third family, NLP, combines classical language analysis and statistical approaches [17] in order to provide very specific solutions that aim to tackle the problem of inferring the true meaning from text. A special attention is paid to linguistic annotations, *Treebanks* [22], [13], [11], [19], which intervene in part-of-speech (POS) tagging, syntactic parsing, morphological analysis and word sense disambiguation. NLP is used in topic modelling and it can be also considered as a pre-processing step for the other two families to provide more meaningful topic inference in terms of semantics. Due to space limitation this family will not be covered, but will be presented in future work.

Given that first two families of methods above come from very different origins (multivariate analysis, Bayesian analysis and AI, respectively), our goal is to present a unified notation and make it homogeneous over different techniques and authors. A general formalization contributes to a better understanding of the relationships between those families of methods and their common properties. This allows to have a common language for the existing state-of-the-art literature considerably simplifies the task of the researcher to identify the reasons of different results obtained with the different techniques over the same set of texts.

The structure of the paper is as follows. First, in Section §2, it is presented the proposed common structures and notation. Then, Factorial methods are presented in Section §3 along with their applications and extensions in the topic modelling domain. Then, Probabilistic PCA and Probabilistic Topic Modelling framework are presented in Sections §4 and §5, respectively. Finally, Section §6 contains a brief discussion among the different methods, conclusions and future work.

2. Methodology: a common notation framework for textual analysis methods

In this section the symbology associated to the different elements appearing in the presented methods is designed and it will be used in the methods presented in the following sections.

2.1. Numerization of the corpus

It appears to be a basic and very early operation in most of the methods from Factorial and Probabilistic families and it consists in representing a set of documents through numeric matrices that represent in each row the distribution of words in one document.

Given a set of documents $\mathscr{D}$ of size $n_{\mathscr{D}}$ and a set of terms $\mathscr{T}$ of size $n_{\mathscr{T}}$, a document is a sequence of words such that for each document $d_j \in \mathscr{D}$ with $j = 1 \dots n_{\mathscr{D}}$, $d_j = (w_1, w_2, \dots, w_{n_{d_j}})$ where n_{d_j} is the number of words in the document d_j and $w_{\wp} \in \mathscr{T}$ with $\wp = 1 \dots n_{d_j}$.

The numerization of the corpus produces a matrix X of dimensionality $n_{\mathscr{T}} \times n_{\mathscr{D}}$, as shown below, where the rows correspond to terms $t \in \mathscr{T}$ (i.e. vocabulary used in the corpus $\mathscr{D}$), and columns correspond to documents $d \in \mathscr{D}$ of the given corpus. Therefore, the number of rows of X is the cardinal of the given vocabulary ($n_{\mathscr{T}}$) and the number of columns is the cardinal of $\mathscr{D}$, $n_{\mathscr{D}}$. Each cell (i, j) is the raw count of n_{ij} occurrences of the term t_i in the document d_j from the collection.

$$X = \begin{array}{cc} & \begin{array}{ccc} 1 & \dots & n_{\mathscr{D}} \end{array} \\ \begin{array}{c} 1 \\ \\ \vdots \\ \\ n_{\mathscr{T}} \end{array} & \begin{bmatrix} & & \\ & \vdots & \\ \cdots & n_{ij} & \cdots \\ & \vdots & \\ & & \end{bmatrix} \end{array} \tag{1}$$

Matrix X is often known as TDM, a term-document matrix, successfully applied in the field of information retrieval [2] and tf-idf computation [26].

Along this paper we will name as x^i the vector $x^i = (n_{i1}, \dots, n_{in_{\mathscr{D}}})$, which represents the profile of a certain term in a corpus, that is, the distribution of the occurrences of term t_i in the documents of $\mathscr{D}$.

2.2. Binarization of documents

An alternative common representation of the corpus is the *Binarization* of documents. It consists in representing each single document $d_j \in \mathscr{D}$ as a binary matrix, $d^{(j)}$, describing the distribution of terms in the document.

Provided that n_{d_j} is the length of the document d_j, the matrix $d^{(j)}$, of dimensionality $n_{d_j} \times n_{\mathscr{T}}$, has $n_{\mathscr{T}}$ terms in columns and n_{d_j} rows representing the positions where terms can be placed along the document.$\wp = 1 \dots n_{d_j}$ indexes the positions of the terms that appear in the document d_j.

$$d^{(j)} = \begin{array}{cc} & \begin{array}{ccc} 1 & \dots & n_{\mathscr{T}} \end{array} \\ \begin{array}{c} 1 \\ \\ \vdots \\ \\ n_{d_j} \end{array} & \begin{bmatrix} & \vdots & \\ \cdots & d^{(j)}_{\wp i} & \cdots \\ & \vdots & \end{bmatrix} \\ & \begin{bmatrix} n_{1j} & \dots & n_{\mathscr{T}j} \end{bmatrix} \end{array} \tag{2}$$

The cell, $d^{(j)}_{\wp i}$, of the binary matrix, $d^{(j)}$, is defined as follows:

$$d^{(j)}_{\wp i} = \begin{cases} 1, & \text{if term } t_i \text{ appears at position} \wp \text{ of the document } d_j \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

3. Factorial methods

3.1. Latent Semantic Analysis

The main goal of LSA [9] is to infer meaningful semantic structures of the terms in documents and to discard those attributable to noise. The internals of LSA reside in the two-way factor analysis using Singular Value Decomposition (SVD). Two is referred to the fact that both terms as well as documents are jointly represented in the same factorial space, thus allowing the analysis of relationships between them.

The matrix X, described in Section §2.1, or its weighted version (using tf-idf numerical statistic from [27]), can be decomposed, under SVD, into the product of three matrices as follows:

$$X = \mathscr{V}_{(n_{\mathscr{T}} \times \mathscr{K})} \Lambda^{\frac{1}{2}}_{(\mathscr{K} \times \mathscr{K})} (\mathscr{U}')_{(\mathscr{K} \times n_{\mathscr{D}})} \tag{4}$$

being $\mathscr{V}_{(n_{\mathscr{T}} \times \mathscr{K})}$ a matrix of eigenvectors of XX', $\mathscr{U}_{(n_{\mathscr{D}} \times \mathscr{K})}$ a matrix of eigenvectors of $X'X$, $\Lambda_{(\mathscr{K} \times \mathscr{K})}$ a diagonal matrix of eigenvalues and $\mathscr{K} = \min\{n_{\mathscr{T}}, n_{\mathscr{D}}\}$ the rank of X.

Let u_α be one of the $\mathscr{K}$ eigenvectors of the matrix $\mathscr{U}_{(n_{\mathscr{D}} \times \mathscr{K})}$, and it is a linear combination of the original set of "document-variables". The projection of X over u_α, $\Psi_\alpha = Xu_\alpha$, is α-th principal component of the dataset, that can be thought as concept or topic. The associated eigenvalue $\lambda_{\alpha\alpha}$ measures the quantity of information retained by Ψ_α from the total information contained in X [3].

Joint representation (of terms and documents) onto factorial space is possible due to transition relations between $\mathscr{V}$ and $\mathscr{U}$ [10] and can be represented through *rescaling factor* or *biplot* representation [20].

In LSA several issues were identified: *synonymy*, *polysemy* and rare event detection. Last one tackled very elegantly in Correspondence Analysis [15], [20] by using a χ^2 metrics.

As an extension of LSA and to overcome the lack of context, in [25] authors present Distributional Semantic Model by extending the Vector Space Model representation in which they introduce the co-occurrence of the terms matrix, $C_{(n_{\mathscr{T}} \times n_f)}$ between all $n_{\mathscr{T}}$ terms and n_f pre-defined terms. The maximization expression becomes:

$$\max_{u_\alpha} u'_\alpha (X'C)'(X'C)u_\alpha \text{ s.t. } u'_\alpha u_\alpha = 1 \tag{5}$$

3.2. Archetypal Analysis

Archetypal Analysis (AA) [8] belongs to the same family of optimization problems such as LSA/PCA, *k*-means or NMF. For instance, in [6], authors present a framework to handle multi-document summarization problem. The formulation of the AA as an optimization problem is as follows:

$$\begin{aligned} &\underset{H,W}{min} \quad \|J - H_{(n_{\mathscr{T}} \times K)} W'J\|^2 \\ &s.t. \sum_{k}^{K} h_k^i = 1, h_k^i \geq 0, \forall i \in \{1 \ldots n_{\mathscr{T}}\} \text{ and } \sum_{i}^{n_{\mathscr{T}}} w_k^i = 1, h_k^i \geq 0, \forall k \in \{1 \ldots K\} \end{aligned} \tag{6}$$

where $Y_{(n_{\mathscr{T}} \times K)} = J'W$ is defined as the matrix of K archetypes in columns, which are built as convex combinations of observations. $W_{(n_{\mathscr{T}} \times K)}$ (estimated) determines the convex combination of J such that columns of Y are placed on the convex hull of data J. In turn, the observations can be approximated as convex combinations of archetypes. And matrix H describes convex combination of archetypes to approximate observations, such that $J \approx HY'$, or in other words, H is the weighting matrix that approximates the archetypal space into the transformed design matrix J. In contrast to NMF, AA decomposes the matrix J into sparser stochastic matrices. And the archetypes, the columns of Y, can be interpreted as topics or latent representation of the data.

4. Probabilistic Principal Component Analysis

In standard PCA, the approach is to maximize the projection of the original data space X (the individuals) onto the latent (unknown) factorial space Ψ. But in the probabilistic version, the idea is to first establish the link from latent space Ψ to original data space X and then, the reverse mapping is found by using the posterior distribution by Bayes theorem. A PPCA is a linear Gaussian *latent* variable model [29], [4].

A particular term profile x^i (defined in §2.1) is defined in [29] as stochastic linear combination of its corresponding projection in the latent space (see §3.1), namely ψ^i (is the i-th row of the matrix Ψ), plus a noise term

$$x^i|\psi^i = (\mu + W\psi^i) + \varepsilon^i, \qquad \varepsilon^i \sim \mathscr{N}(0, \sigma^2 I) \tag{7}$$

where μ is the global mean of, $X_{n_{\mathscr{T}} \times n_{\mathscr{D}}}$ and W is the parameter matrix that contains the *factor loadings*. After several simplifications, such as homoschedasticity hypothesis, the posterior distribution of the latent variables, ψ^i, is obtained with the Bayes Law and formulated as:

$$p(\psi^i|x^i) = \mathscr{N}\left(M^{-1}W'(x^i - \mu),\ \sigma^2 M^{-1}\right) \tag{8}$$

where $M_{K\times K} = \sigma^2 I + W'W$. Details have been omitted due to space constraints.

The marginal log-likelihood of the data, X, is formulated as:

$$\mathscr{L}(\mu, \sigma^2, W) = \sum_{i=1}^{n_{\mathscr{T}}} \log\left\{p(x^i)\right\} = -\frac{n_{\mathscr{T}}}{2}\left\{n_{\mathscr{D}} \log(2\pi) + \log|C| + \mathrm{tr}(C^{-1}S)\right\} \tag{9}$$

being $C = WW' + \sigma^2 I$ the model covariance matrix and S the empirical covariance matrix of $X_{n_{\mathscr{T}} \times n_{\mathscr{D}}}$ and $\pi = 3.1415\ldots$ By this technique, it can be obtained the approximation (by EM algorithm) to the same axes as in LSA or PCA.

5. Probabilistic topic modelling

The probabilistic mixture models [24], [4] are characterized by the fact that the data is generated by one of the mixture components. For example, a mixture of Gaussians can approximate any type of continuous distributions, including multimodal [7]. In this work, each mixture component is referred as topic.

Under the assumptions of independence between document size and words' sequence, the documents as an iid sample, and that the document can be expressed as mixture of the set of topics, the generic likelihood is as follows:

$$\begin{aligned}\mathscr{L}(\theta) = \prod_{j=1}^{n_{\mathscr{D}}} \sum_{k=1}^{K} \Bigg(& P(N = n_{d_j}|\mathrm{Z} = z_k \wedge \theta)\cdot \\ & \cdot P\Big(\bigwedge_{\wp=1:n_{d_j}} D_{\wp} = w_{\wp}|\mathrm{Z} = z_k \wedge \theta\Big) P(\mathrm{Z} = z_k|\theta)\Bigg)\end{aligned} \tag{10}$$

where Z is a discrete random variable with values in $\mathscr{Z} = \{z_1, \ldots, z_K\}$ and the associated probability space is defined as follows:

$$\langle \mathscr{Z}, \mathscr{P}(\mathscr{Z}), P_{\mathscr{Z}} \rangle \tag{11}$$

where $\mathscr{Z}$ is the sample space (set of topics), $\mathscr{P}(\mathscr{Z})$ is parts of $\mathscr{Z}$ and $P_{\mathscr{Z}}$ is the probability function associated to $\mathscr{P}(\mathscr{Z})$. $P_{\mathscr{Z}}$ is built on top of $p_{\mathscr{Z}} = P(Z = z_k)$ for $k = 1 \ldots K$, provided that $\mathscr{P}(\mathscr{Z})$ is a σ-algebra. $D_{\wp}$ is a random variable indicating which term is observed in any position $\wp$ of document d_j. And θ is the set of distributional parameters.

5.1. Generative model

In Generative model [23] the probability of a word remains constant for all documents in a corpus and independent of the words in other positions of same document, as well as independent of the position where it is observed, conditioned on the topic and a set of parameters. Therefore,

$$D_{\wp} | Z = z_k \wedge \theta \sim \mathrm{Cat}(\pi_{1k}, \ldots \pi_{n_{\mathscr{T}}k}) \tag{12}$$

where π_{ik} is the probability of occurrence of term $t_i \in \mathscr{T}$ given topic $Z = z_k$. Under this approach, the likelihood function in (10) is reformulated as follows:

$$\mathscr{L}(\theta) = \prod_{j=1}^{n_{\mathscr{D}}} \sum_{k=1}^{K} \left(P(N = n_{d_j} | Z = z_k \wedge \theta) \left(\prod_{i=1}^{n_{\mathscr{T}}} \pi_{ik}^{n_{ij}} \right) \cdot P(Z = z_k | \theta) \right) \tag{13}$$

Similarly, the likelihood function for the Multinomial model can be rewritten as follows:

$$\mathscr{L}(\theta) = \prod_{j=1}^{n_{\mathscr{D}}} \sum_{k=1}^{K} \left[P(N = n_{d_j} | Z = z_k \wedge \theta) \left(\frac{n_{d_j}!}{\prod_{i=1}^{n_{\mathscr{T}}} n_{ij}!} \prod_{i=1}^{n_{\mathscr{T}}} \pi_{ik}^{n_{ij}} \right) P(Z = z_k | \theta) \right] \tag{14}$$

And the likelihood function for the Multivariate Bernoulli model is as follows:

$$\mathscr{L}(\theta) = \prod_{j=1}^{n_{\mathscr{D}}} \sum_{k=1}^{K} \left[\left(\prod_{i=1}^{n_{\mathscr{T}}} \left(x_j^i \pi_{ik} + (1 - x_j^i)(1 - \pi_{ik}) \right) \right) P(Z = z_k | \theta) \right] \tag{15}$$

Having, for the given document d_j, the realization of the variable X_j^i is $x_j^i \in \{0, 1\}$, which states whether the term t_i is present in the specific document d_j or not.

5.2. Latent Dirichlet Allocation

The Latent Dirichlet Allocation (LDA) is a bayesian hierarchical model [5]. As opposite to models presented in Section §5, the documents are associated with multiple topics simultaneously [5]. Hence, the random vector D (defined in Section §5) with the sequence of words in the document, jointly appears with the random vector $\mathbb{Z}$ (sequence of topics in the document):

$$P(D=d \wedge \mathbb{Z}=(z_1,z_2,\ldots,z_{n_d})) = P(\bigwedge_{\wp=1:n_d} (D_{\wp}=w_{\wp} \wedge Z_{\wp}=z_{\wp})) \tag{16}$$

The likelihood function of the parameters α and θ in the corpus $\mathscr{D}$ can be defined as follows:

$$\begin{aligned} \mathscr{L}(\alpha,\theta) &= \prod_{j=1}^{n_{\mathscr{D}}} P(D=d_j|\alpha \wedge \theta) \\ &= \prod_{j=1}^{n_{\mathscr{D}}} \int P(\zeta_j|\alpha)\left(\prod_{\wp=1}^{n_{d_j}} \sum_{k=1}^{K} P(D_{\wp}=w_{\wp}|Z_{\wp}=z_k \wedge \theta)P(Z_{\wp}=z_k|\zeta_j)\right) d\zeta_j \end{aligned} \tag{17}$$

6. Conclusion

Although multivariate and probabilistic topic modelling families are very different approaches from the conceptual point of view, this work shows that using common notation, commonalities and differences among them can be analysed in detail.

In the multivariate setting no distributional assumptions are made, but it provides a very clear interpretation and geometrical representation of the associations between the topics, documents and terms, so that visual inspection of the results can provide a global overview of the interactions among these. These methods optimize a function related with the information of the original data set: residual sum of squares for AA, and maximize projected variance for PCA. In general, the optimal projection directions are found based on diagonalization techniques applied to combinations of the TDM or TSM.

The probabilistic methods, on the other side, do not provide geometric representation, but are more flexible while capturing associations between topics, documents and terms. Also, they assume a predetermined number of topics from the beginning, while the multivariate methods allow the determination of the relevant topics as an output, by analyzing the quantity of information preserved in each of the topics and keeping the significant ones. The flexibility of LDA, allowing every word in a document to be associated with a different topic, does not seem very realistic either.

The PPCA looks like a very interesting method as a combination of probabilistic and multivariate methods, nevertheless the Gaussian assumption does not correspond to the distribution of the terms in the document.

Therefore, although being a simple linear models, multivariate models look like the most conservative modelling schema as they do not make any distributional assumptions and the interpretation of the results is straightforward.

However, what this analysis is making evident is that all of the proposed methods provide elements to characterize the topics in terms of the documents in the topic, or the words more representative of the topic, but all of them rely on the comprehension of which topics really are, to the interpretational habilities of the analyst, thus pointing to a missing final step in the topic modelling research field which is to provide a concept (or a label) for each of the discovered topics.

Ongoing research is the generalization of the proposed common formal framework to include the main concepts of the NLP methods. And once the common framework has been established, the next step will be to apply the proposals of already existing and new methodologies on testing real cases. Also, the incorporation of Natural Language Processing into the pipeline of LSA or similar techniques, opens the door to introduce inductive reasoning and ontologies of words and terms to apply inductive learning principles to the extraction of explicit concepts associated to the discovered topics, so that a proposal for automatic interpretation of topics can be formalized.

References

[1] Parvin Ahmadi, Iman Gholampour, and Mahmoud Tabandeh. Cluster-based sparse topical coding for topic mining and document clustering. *ADAC*, 12(3):537–558, Sep 2018.

[2] Ricardo Baeza-Yates, Berthier Ribeiro-Neto, et al. *Modern information retrieval*, volume 463. ACM press New York, 1999.

[3] Jean-Paul Benzécri et al. Lanalyse des donnees, tome ii. *Lanalyse des correspondances. Dunod Press, Paris*, 1973.

[4] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

[5] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.

[6] Ercan Canhasi and Igor Kononenko. Multi-document summarization via archetypal analysis of the content-graph joint model. *Knowledge and Information Systems*, 41(3):821–842, Dec 2014.

[7] M. A. Carreira-Perpinan. Mode-finding for mixtures of gaussian distributions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1318–1323, Nov 2000.

[8] Adele Cutler and Leo Breiman. Archetypal analysis. *Technometrics*, 36(4):338–347, 1994.

[9] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal Of The American Society For Information Science*, 41(6):391–407, 1990.

[10] Luc Devroye, Ludovic Lebart, A Morineau, and J.-P Fenelon. Tratement des donnees statistiques: Methodes et programmes. *Journal of the American Statistical Association*, 75:1040, 12 1980.

[11] Christiane Fellbaum. *WordNet: An Electronic Lexical Database*. Bradford Books, 1998.

[12] Marti Font, Xavier Puig, and Josep Ginebra. Bayesian Analysis of the Heterogeneity of Literary Style. *Revista Colombiana de Estadstica*, 39(2):205–227, 2016.

[13] W. N. Francis and H. Kucera. Brown corpus manual. Technical report, Department of Linguistics, Brown University, Providence, Rhode Island, US, 1979.

[14] Wolfgang Gaul and Dominique Vincent. Evaluation of the evolution of relationships between topics over time. *Advances in Data Analysis and Classification*, 11(1):159–178, Mar 2017.

[15] Michael J. Greenacre. Correspondence analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(5):613–619, 2010.

[16] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, pages 50–57, New York, NY, USA, 1999. ACM.

[17] Nitin Indurkhya and Fred J Damerau. *Handbook of natural language processing*, volume 2. CRC Press, 2010.

[18] Serge Iovleff. Probabilistic auto-associative models and semi-linear pca. *Advances in Data Analysis and Classification*, 9(3):267–286, Sep 2015.

[19] S. Johansson. *Manual of Information to Accompany the Lancaster-Oslo/Bergen Corpus of British English, for Use with Digital Computers*. ICAME collection of English language corpora. Univ., Department of English, 1978.

[20] I.T. Jolliffe. *Principal Component Analysis*. Springer Series in Statistics. Springer, 2002.

[21] Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In *Proceedings of the 13th International Conference on Neural Information Processing Systems*, NIPS'00, pages 535–541, Cambridge, MA, USA, 2000. MIT Press.

[22] Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. Building a large annotated corpus of english: The penn treebank. *Comput. Linguist.*, 19(2):313–330, June 1993.

[23] Kamal Nigam, Andrew Kachites Mccallum, Sebastian Thrun, and Tom Mitchell. Text classification from labeled and unlabeled documents using em. *Machine Learning*, 39(2):103–134, May 2000.

[24] D. Peña. *Análisis de datos multivariantes*. Mc Graw Hill, 2002.

[25] Martin Rajman and Romaric Besançon. Stochastic distributional models for textual information retrieval. *Proc. of 9th ASMDA*, pages 80–85, 1999.

[26] G. Salton and M.J. McGill. *Introduction to modern information retrieval*. McGraw-Hill computer science series. McGraw-Hill, 1983.

[27] Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. *Introduction to information retrieval*, volume 39. Cambridge University Press, 2008.

[28] Dawit G. Tadesse and Mark Carpenter. A method for selecting the relevant dimensions for high-dimensional classification in singular vector spaces. *Advances in Data Analysis and Classification*, Jan 2018.

[29] M. E. Tipping and Christopher Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society, Series B*, 21/3:611622, January 1999.

[30] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 2(1):37 – 52, 1987.

[31] Yuhan Zhang and Haiping Xu. Sltm: A sentence level topic model for analysis of online reviews. In *SEKE*, pages 449–453, 2016.

This page intentionally left blank

Data Science and Decision Support Systems

This page intentionally left blank

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210128

The Emotional Impact of Hotel Room Colour in Spanish and Equatorian Tourists

Mónica P. BUENAÑO[a], Lledó MUSEROS[b] and Luis GONZALEZ-ABRIL[c]
[a]*Universidad Técnica del Norte,* [b]*Universitat Jaume I,*
[c]*Universidad de Sevilla*

Abstract. Research suggests that colour plays an important role in creating wellness emotions in hotel customers. This paper considers that tourists' needs for wellness may be satisfied by manipulating existing elements of a hotel, such as the colour of a hotel room. The paper studies the relationship between tourists' emotions and the main colour of a hotel room, and also the relationship between that emotion and their intention to stay in the hotel, and even the price that the tourists are willing to pay. Also, the paper studies the role of cultural differences in these relationships, specifically between Spanish and Equatorian tourists.

Keywords: colour and emotions, touristic image, tourism marketing, recommender system, data science.

Introduction

Colour is considered an indispensable element in interior design [1]. It allows designers to alter the characteristics of the environment in an easy and economical way [2], and to create desirable and pleasant environments [3]. It has a strong visual component which is able to attract the client's attention [4]. Furthermore, it is a subtle stimulant with a significant impact on people's emotions and behavior [5-7] , and it can increase motivation and well-being [8]. Therefore, colour selection for a hotel room decoration is very important in order to design a comfortable and wellness-oriented environment. In some ways, colour selection of the decoration of a room can guarantee an affective commitment between the client and the hotel [9]. Moreover, these aspects can influence the choice to stay in a hotel, which is a decision made not only on a rational basis but also under emotional and wellness-oriented considerations [10].

The new trends in the demand of hotels in the wellness sector have a direct influence in the generation of new products and services in tourism and hospitality [11]. Luxury hotels with spa facilities are recognized as the leading providers of wellness services [11], but there are also smaller hotels that cannot afford spa and high-cost facilities, but can offer wellness services by finding strategies to meet the needs of wellness travelers and that do not demand large economic investments. In this way, Thus, Dimitrovski & Todorović [12] propose that wellness tourism should be oriented towards the emotional motivation of the guest instead of using luxurious spas, taking into account that emotion is a key point in the evaluation of personal well-being [13]. Thus, the paper presented in this paper proposes the study of the emotional link that the colour of a room can generate in the guests as a tool for small hotels to improve their wellness image.

Research on the emotional impact of colour and behavioral intentions has been developed in different areas. For example, it has been shown that colour on food labels influences emotions, selection and intention to pay more for the product [14]. Wu, Li & Liu [15] showed that there is a close relationship between the colour of food packages, emotions and user satisfaction. It was also found that the colour of the office and study environment influences emotions, work and learning performance [16].

Approaches in the tourism and hospitality field have studied the relationship between the emotional response to the colour of a restaurant and the intention to choose it and enter [17]. Also, the works presented in [18] demonstrate that guests who reported emotions of comfort, well-being and relaxation can build a relationship of loyalty and intention to pay more for the service. Haller [19] mentions that colour is the first thing that is perceived when entering a room. It can influence behavior, feelings, such as feeling more relaxed, eating more and spending more. Countryman & Jang [6] demonstrated that colours are the most influential element to affect overall hotel lobby impression. The approaches in [13] specify that guests prefer blue or green as colours for the rooms, since they find these colours more relaxing, and calming and therefore better rooms for having a good quality of sleep and well- being. In fact, these colours are used in luxury hotels to induce pleasure [20]. On the contrary, the colour red has been associated with high levels of arousal and restlessness [21] and therefore it is not recommended as the colour of a hotel room.

Therefore, it is important to study the link between colour, emotion and behavior. To understand this relationship, Mehrabian and Russell SOR model [22] affirms that the physical environment acts as stimuli (S) that lead to people's emotions (O), which in turn, drive behavioral responses (R).

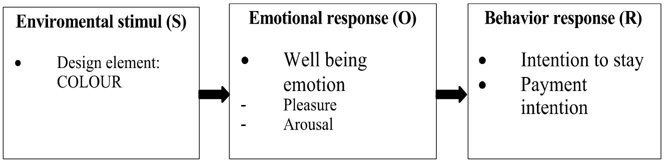


Figure 1. SOR model, relationship between colour stimuli, emotions and behavioral responses

To assess emotional responses to environmental stimuli, Russell and Lanius [23] propose a model (PA model) concerning affective appraisals of environments. It is based on two bipolar dimensions (Pleasantness – unpleasantness; Arousing – not arousing) (Figure 2).

Therefore, this study suggests that through the manipulation of colour in different elements of the physical environment of a hotel room, emotions of well-being can be created, which in turn, drive the intention to stay and the price that one would be willing to pay (Figure 1).

The present study also considers exploring cultural influence in the link between colour and emotions, since emotional responses to colour can be determined by cultural traits. Thus, in this paper the following hypothesis are going to be explored:

H1: The intention to stay is related to colours.

H2: Changes to the colour of the hotel room generate different emotions.

H3: The intention to stay in the hotel is related to positive emotions.
H4: The price the tourists are willing to pay for a room is related to their nationality.

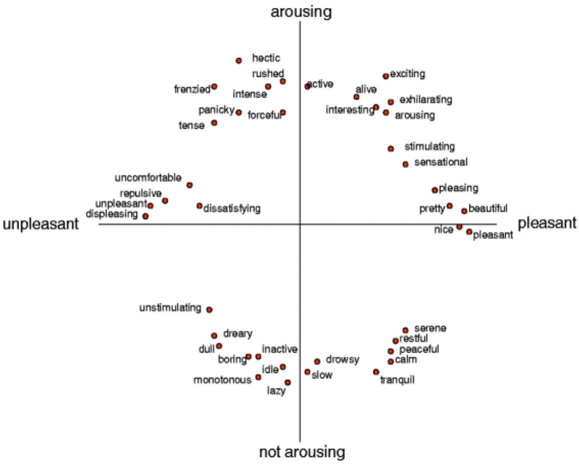


Figure 2. Categorical affective descriptors located in the space for the PA model. Adapted from Russell & Lanius, 1984

The rest of the paper is organized as follows, first a brief overview of the QCD model used in this approach is presented, then the survey designed for data collection is described. Section 4 analyzes the results collected with the survey to verify the hypothesis defined. Finally, conclusions and future work are presented.

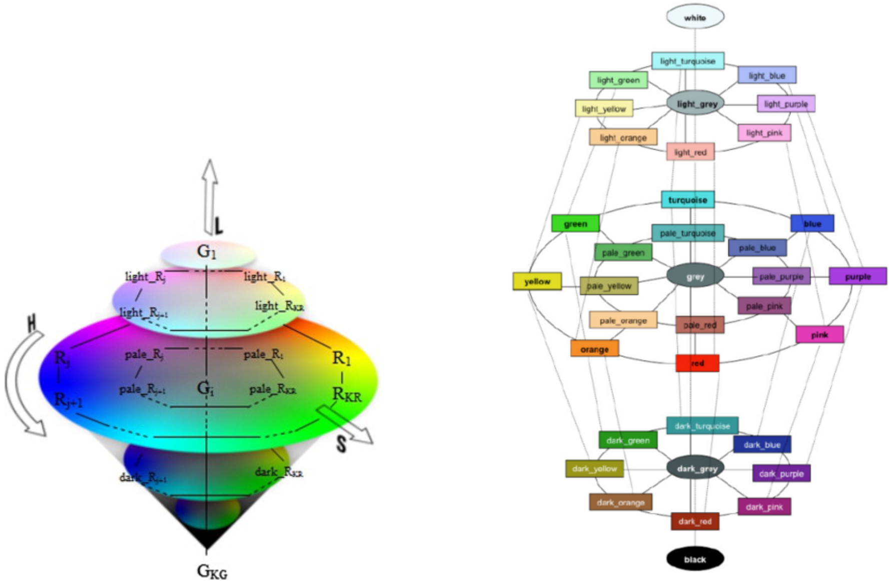
1. Overview of the Qualitative Colour Descriptor (QCD)

The QCD model [24] defines a reference system in the HSL colour space for qualitative colour description, which is built according to Figure 3 and defined as:

$$QC_{RS} = \{uH, uS, uL, QC_{NAME1..5}, QC_{INT1..5}\}$$

where uH is the unit of Hue; uS is the unit of Saturation; uL is the unit of Lightness; $QC_{NAME1..5}$ refers to the colour names; and $QC_{INT1..5}$ refers to the intervals of HSL coordinates associated with each colour. The chosen QC_{NAME} and QC_{INT} are shown in Figure 3.b

The QCD model has a relational structure and it can be organized in a conceptual neighborhood diagram (CND) [25] according to how a colour can be transformed into another by changing its luminosity, saturation or hue. A CND for the computational QCD is built and shown in Figure 3(b).



(a) (b)

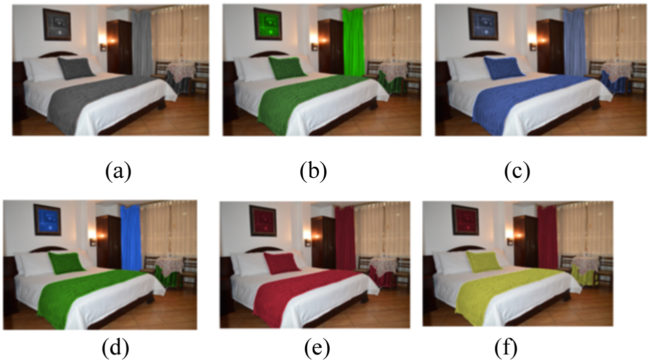
Figure 3. Diagram for describing QCD: (a) Discretization of the HSL colour space; (b) CND corresponding to QCD.

2. Design of the Experiment

An online survey (Figure 5) was designed using the LimeSurvey[1] software. Using this tool, a survey was designed where 6 demographic questions were defined (nationality, gender, age, level of education, favourite colour, and number of nights in a hotel during last year). Then the survey presents a set of 9 images of a hotel room (see Figure 4 for examples). There were 3 different types of questions for each image (named *base questions*). The first question was aimed at determining the relationship between the tourist's emotion and the color of a hotel room, the second question asked for the conative aspect of the tourist (their intention to stay in such a room as the one in the image, it was rated using a five-point Likert scale), and finally the survey asked about the price that the tourist would agree to pay for the room. The first set of questions were based on a basic room at the Hotel Plaza Victoria in Ecuador (Figure 4a), with gray decoration. This room is used also at the end of the test as a control question. Then the colours of decorative elements (group 1: curtain and headboard, group 2: the cushion, footboard, and tablecloth) in the room were colorized using the qualitative colours defined in the QCD model [25]. Following the approaches [13-14], and in order to verify that green and blue colours are prefered by the guests over red colours, the colours green, blue, red and yellow have been chosen for the decoration of the decorative elements of the room. Each set of base questions was made for two rooms decorated with colours with the symbol green (as green, pale green, dark green, or light green) in its label (as in the example in Figure 4b), secondly for two rooms with any blue of the QCD decoration (Figure 4c), then a room with any green an any blue decoration (Figure 4d), then a room

[1] https://www.limesurvey.org/es/.

with only yellow or red decorations (Figure 4e), and finally a room with red and yellow decoration was presented (Figure 4f). Not all the tourists saw the same image, since the colours of the room were randomly generated using the QCD model. This means that, for instance, the blue colour rooms could present different RGBs colours if the corresponding colour label inside the QCD model had the blue, pale-blue, light-blue or dark-blue label. The same happens with the rest of colours. Therefore, each tourist had to answer the 3 base questions related to 9 images of the same hotel room, but with a different colour. The same room image was used to avoid participant fatigue, and it was verified with the members of the research team that the survey was carried out quickly and without fatigue.


(a) (b) (c)

(d) (e) (f)

Figure 4. Examples of images of a hotel room used in the survey.

3. Data Collection and Analysis

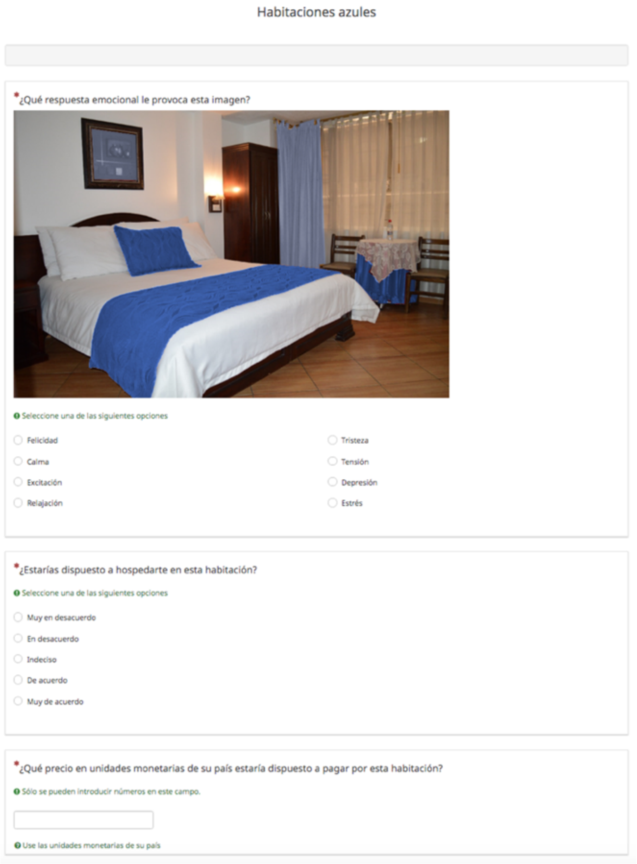
The experiment was carried out with the collaboration of 239 tourists, mainly Spaniards and Ecuadorians. Analyzing the data collected with the survey, the tourists answering 51.9% were female, 36.4% participants were male, and 11.7% preferred not to say their gender; the age of participants ranged from 18 to 60 years, and 38.9% had graduate degrees and 38.9% had postgraduate degrees. The majority (70%) of the participants spent 15 or fewer days a year in a hotel room.

Next, the collected results are analyzed in relation with the initial hypothesis defined.

H1: The intention to stay is related to the colours.

To examine the effects of colours on the intention to stay at the hotel, tourists were asked first to evaluate a room hotel with gray decoration as our basis (gray scale or neutral colour), and 62% of them answered a positive intention to stay in the hotel (see Table 1). This shows that upon a first observation, the gray decoration is accepted by the tourists. Also gray has been chosen as the baseline colour because it is considered a neutral and practical hue, described as serious or stately and associated with official buildings [26].

Then, by varying the colours of the room, and presenting seven random rooms with different colour decorations to each of the 239 users, 1673 responses were obtained (see Table 1). The results shows that the intention to stay in the same room was 44%, which is lower than the baseline, and indicates that not all colours are suitable for the interior of a room and may have a negative effect. Therefore, due to the difference between these values (44% and 62%), the hypothesis is confirmed: the colour of a room significantly influences the intention to stay.



Habitaciones azules

*¿Qué respuesta emocional le provoca esta imagen?

Seleccione una de las siguientes opciones

○ Felicidad ○ Tristeza
○ Calma ○ Tensión
○ Excitación ○ Depresión
○ Relajación ○ Estrés

*¿Estarías dispuesto a hospedarte en esta habitación?

Seleccione una de las siguientes opciones

○ Muy en desacuerdo
○ En desacuerdo
○ Indeciso
○ De acuerdo
○ Muy de acuerdo

*¿Qué precio en unidades monetarias de su país estaría dispuesto a pagar por esta habitación?

Sólo se pueden introducir números en este campo.

Use las unidades monetarias de su país

Figure 5. Example of the survey and the base questions for a blue room.

H2: Changes to the colour of the hotel room generate different emotions.

Table 1. Intention to stay and colour

	Gray		**Colour**	
	Frecuency	%	Frecuency	%
Doubt	057	24	396	24
No	033	14	541	32
Yes	149	**62**	736	**44**
Total	239	100	1673	100

To examine the effect of the colour change to the base room on the emotional responses based on the PA model, tourists were asked to rate randomly generated room images with combinations of blue, green, red, and yellow colours. A contrast of proportions is used in order to test the hypothesis that changes in the colour of the decoration of the rooms significantly affect and change the emotions of the tourists. For this, if the adjective of the gray room is not changed, the variable takes the value 0 and otherwise 1 (see Table 2). Thus, if colour does not affect emotion, the probability (p) to change the emotions is zero, that is $p = 0$, nevertheless the hypothesis is relaxed and the null hypothesis is H0: "Colour does not affect emotions ($p \leq 0.2$)".

The results of the contrast of proportions showed that the *p*-value is close to zero; therefore, the null hypothesis is rejected and hence, it can be concluded that changes in the colour of the room decoration change the emotions. It is important to note that 86% of responses change the emotion with respect to the gray room.

Table 2. Frequency of change of emotion in relation to different colour decorations for a hotel room.

Change emotion	Frecuency	%
0	242	14%
1	1431	86%
Total	1673	100%

H3: The intention to stay in the hotel is related to positive emotions

From the responses of the participants, a new variable called “positive emotion” has been created as follows: if the emotions are “Calm”, “Happiness”, “Relaxation”, the positive emotion is 1; if the emotions are “Depression”, “Stress”, “Tension” and “Sadness” the positive emotion is -1; and if the emotion is “Exciting”, then the positive emotion is 0.

Hence, the contingency table between this new variable is given in Table 3. A Chi square test was applied to confirm the relationship between positive emotions and the intention to stay in the hotel and the *p*-value of this test was 1.17 e-230, which indicates that the two variables are related.

H4: The price he/she are willing to pay for a room is related to the nationality.

A box plot comparing the price that Spanish and Ecuadorian tourists are willing to pay for a room is shown in Figure 6. To this, all the currency units have been converted to euros[2] in order to be able to compare the values. It can be seen in Figure 3 that Spaniards are willing to pay more for the room than Ecuadorians.

Table 3. Intention to stay

		Doubt	No	Yes	All
Positive emotion	-1	246	496	35	777
	0	62	22	88	172
	1	88	23	613	724
	All	396	541	736	1673

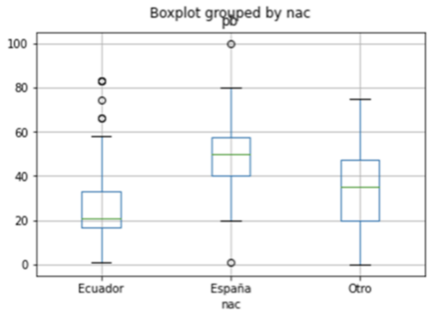


Figure 6. The price that the tourists are willing to pay

[2] Change of day 30/04/2021

Table 4 shows that on average Spaniards would be willing to pay 49.56 euros, while Ecuadorians would pay 23.62 euros less than Spaniards. This is natural due to the fact that Spaniards have greater purchasing power. This was confirmed by Durán-Román, Cárdenas-García, & Pulido-Fernández [27], that the willingness to pay to improve the experience in the destination has a direct relationship with the income and budget of the tourists visiting Andalucía. Although income allows paying more for a service, other studies found that nationality had a more significant effect on willingness to pay for some tourism activity than other factors [28], even the foreign visitors' willingness to pay is twice that of domestic visitors' willingness to pay [29].

In order to test the hypothesis, *the Mann Whitney Test* (test of difference of mean) has been used and a *p*-value of 8.28 e-14 is obtained. Therefore, there is significative difference between the mean values willing to pay for a room between Ecuadorians and Spaniards.

Table 4. Average willingness to pay for a room

Nacionality\ Price (€)	**Count**	**Mean**	**Standard deviation**
Ecuador	177.0	25.94	15.16
Spain	43.0	49.56	18.09
Other	19.0	32.423	21.796

4. Conclusions and Future Work

This study studied the relation between the colour of the decoration of the room in a hotel and the tourists' emotions. Also, it studied if the colour decoration of the rooms affects tourists' intention to stay in the hotel, and finally it studied the price that tourists are willing to pay for a room and the differences between different cultures: Spanish and Ecuadorian tourists. The findings of the study point out that the colour of a room significantly influences the intention to stay and the emotions evoked by the room. Moreover, the results showed that not all colours are suitable for the interior decoration of a room; therefore, the incorrect selection of the colour for that decoration can affect the tourists' emotions and the intention to stay in the hotel. This reinforces the finding showed in Tantanatewin & Inkarojrit [33], that expressed that interior design, with colour as the most influential factor, influences the behavior of people.

The findings from this study also show that there is a relationship between positive emotions and the intention to stay in the hotel. And finally, regarding the willingness to pay for a hotel room, the results of this study confirm the initial hypothesis, which established that the price that the tourists are willing to pay for a room is related to their nationality. Our findings indicate that Spaniards are willing to pay more for a room than Ecuadorians.

In general, understanding the link between colour, emotion and behavior can guide hotel managers in selecting the appropriate colour to strategically manipulate guests' emotions and an affective commitment between the client and the hotel. Therefore, the results of this study are important in order to create a recommender system for hotel marketing materials. The future recommender system is intended to be used to raise the impact of the marketing materials among prospective tourists and then improve the touristic image of the hotel. In this way, the recommender system can be also used to design customized materials for different types of clients.

As future work, the specific colours that enhance tourists' perceptions and emotions and generate a positive emotion increasing their intention to stay in the hotel will be studied. Also, the study will be complemented with the analysis of the mood of a person and its effect on the emotion and behavior.

Acknowledgments

This project has been funded by the Ministry of Economy and Commerce with project contract TIN2016-88835-RET and by the Universitat Jaume I with project contract UJI-B2020-15. Also, this research has been partially supported by the EDITH Research Project (PGC2018-102145-B-C21,C22 (AEI/FEDER, UE)), funded by the Spanish Ministry of Science, Innovation and Universities.

References

[1] D. Oner Ozdas and M. Kazak, Colour preference between adults and children during a dental treatment session, *Physiol. Behav., vol. 169*, pp. 165–168, 2017.

[2] L. Sliburyte and I. Skeryte, What we know about consumers' color perception, *Procedia - Soc. Behav. Sci., vol. 156,* pp. 468–472, 2014.

[3] M. Heide, K. Laerdal, and K. Grønhaug, The design and management of ambience-Implications for hotel architecture and service, *Tour. Manag., vol. 28, no. 5,* pp. 1315–1325, 2007.

[4] J. A. Bellizzi and R. E. Hite, Environmental Color , Consumer Feelings , and Purchase Likelihood, *Psychol. Mark., vol. 9, no. 5*, pp. 347–363, 1992.

[5] J. Barsky and L. Nash, Evoking emotion: Affective keys to hotel loyalty, *Cornell Hotel Restaur. Adm. Q., vol. 43, no. 1,* pp. 39–46, 2002.

[6] C. C. Countryman and S. Jang, The effects of atmospheric elements on customer impression : the case of hotel lobbies, *Int. J. Contemp. Hosp. Manag., vol. 18, no. 7,* pp. 534–545, 2006.

[7] S. Han, D. Choi, and Y. J. Cha, The Effect of Colour on the Anchoring Heuristic in Consumer Decision Making, *J. Eur. Psychol. Students, vol. 3, no. 10,* pp. 19–27, 2014.

[8] C. Yu and H. Yoon, The role of colour in 'health and wellbeing' of the built environment, *Indoor Built Environ., vol. 19, no. 4,* pp. 403–404, 2010.

[9] H. Choi and J. Kandampully, The effect of atmosphere on customer engagement in upscale hotels: An application of S-O-R paradigm, *Int. J. Hosp. Manag., no. June*, 2018.

[10] J. Cheng, T. Tang, H. Shih, and T. Wang, Designing lifestyle hotels, I*nt. J. Hosp. Manag.*, vol. 58, pp. 95–106, 2016.

[11] H. Han, K. Kiatkawsin, H. Jung, and W. Kim, The role of wellness spa tourism performance in building destination loyalty: the case of Thailand, *J. Travel Tour. Mark., vol. 8408,* pp. 1–16, 2017.

[12] D. Dimitrovski and A. Todorović, Clustering wellness tourists in spa environment, *Tour. Manag. Perspect., vol. 16,* pp. 259–265, 2015.

[13] A. Hee Lee, B. Denizci Guillet, and R. Law, Tourists' emotional wellness and hotel room colour, *Curr. Issues Tour.,* pp. 1–7, 2016.

[14] M. Shen and Z. Gao, Blue or red? how color affects consumer information processing in food choice, in *Agricultural & Applied Economics Association Annual Meeting*, 2016, pp. 1–14.

[15] T. Wu, Y. Li, and Y. Liu, *Study of color emotion impact on leisure food package design, vol. 714, 2017*, pp. 612–619.

[16] N. Savavibool, The Effects of Colour in Work Environment: A systematic review, *Environ. Proc. J., vol. 1, no. 4,* p. 262, 2016.

[17] W. Tantanatewin and V. Inkarojrit, The influence of emotional response to interior color on restaurant entry decision, *Int. J. Hosp. Manag., vol. 69*, pp. 124–131, 2018.

[18] A. Wong, The role of emotional satisfaction in service encounters, *Manag. Serv. Qual. An Int. J.*, vol. 14, no. 5, pp. 365–376, 2004.

[19] K. Haller, Colour in interior design, in *Colour Design: Theories and Applications*, Woodhead Publishing Limited, 2012, pp. 551–584.

[20] D. Kim, H. Hyun, and J. Park, The effect of interior color on customers' aesthetic perception, emotion, and behavior in the luxury service, *J. Retail. Consum. Serv.*, vol. 57, no. July, p. 102252, 2020.

[21] C. Jin, H. Noguchi, J. Qiu, H. Wang, Y. Sun, and Y. Lin, The effect of color light combination on preference for living room, in *12th China International Forum on Solid State Lighting* (SSLCHINA), 2015, pp. 139–142.

[22] A. Mehrabian and J. Russell, *An approach to environmental psychology*, Cambridge, MAMIT Press, 1974.

[23] J. A. Russell and U. F. Lanius, Adaptation level and the affective appraisal of environments, *J. Environ. Psychol*., vol. 4, no. 2, pp. 119–135, 1984.

[24] Z. Falomir, L. Museros, and L. Gonzalez-Abril, A model for colour naming and comparing based on conceptual neighbourhood. An application for comparing art compositions, *Knowledge-Based Syst., vol. 81*, pp. 1–21, 2015.

[25] Freksa C. Spatial computing. Cognitive and Linguistic Aspects of Geographic Space: New Perspectives on Geographic Information Research. In: *Raubal M, Mark DM, and Frank AU, editors*. Berlin: Springer Berlin Heidelberg. p. 23–42, 2013.

[26] S. Kurt and K. K. Osueke, The effects of color on the moods of college students, *SAGE Open*, vol. 4, no. 1, p. 215824401452542, 2014.

[27] J. L. Durán-Román, P. J. Cárdenas-García, and J. I. Pulido-Fernández, “Tourists’ willingness to pay to improve sustainability and experience at destination,” J. Destin. Mark. Manag., vol. 19, no. January, 2021.

[28] M. Reynisdottir, H. Song, and J. Agrusa, “Willingness to pay entrance fees to natural attractions: An Icelandic case study,” Tour. Manag., vol. 29, no. 6, pp. 1076–1083, 2008.

[29] S. Piriyapada and E. Wang, “Modeling Willingness to Pay for Coastal tourism resource protection in ko Chang Marine National Park, Thailand,” Asia Pacific J. Tour. Res., vol. 20, no. 5, pp. 515–540, 2015.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210129

A Customer Churn Detection Model for the Pay-TV Sector

Vicente LÓPEZ [a], Rebeca EGEA [b] Lledó MUSEROS [a] and Ismael SANZ [a]
[a] *Universidad Jaume I, Spain*
[b] *Mirada TV, Spain*

Abstract. The business environment today is characterized by high competition and saturated markets. Pay-tv platforms there are not an exception. Because of that, the cost to acquire new customers is much higher than the cost of retaining the existing customers. Therefore, it is important for Pay-TV platforms to keep controlled the Customer Churn. Therefore, the paper studies existing models used to predict Customer Churn in other context -like telecommunication companies customer Churn-and adapts them to the Pay-TV context. Another big problem faced in the paper is the fact that, in the data set udes in the paper there are not personal metrics, which are indispensables to solve the problem. Therefore this approach has defined new metrics in order to be able to predict customer churn.

Keywords. Data mining, Imbalance Classification Problem, Customer churn prediction, Pay-TV platforms.

1. Introduction

We live in a competitive world in which most of services companies have to face the problem of customer churn. Pay-TV platforms are not an exception. The competitivity in this field has grown up, and now it is more expensive to get new customers. Losing customers always means a loss of revenue/profit to the company, but if we consider also the growing costs of getting new customers, the loss can be unaffordable for the company and this can lead the company to the bankrupt.

In Pay-TV paradigm, we can adopt the same definition that was given in [1]: ”Churn is defined to be the activity of customers leaving the company and discarding the services offered by it due to dissatisfaction of the services and/or due to better offering from other providers”. In this approach, our goal is to detect customers with high risk of churn in order to be able to take the necessary actions to prevent it.

There are studies about churn in the field of the telecommunication companies [2], e-commerce [3] or even general studies to solve the problem in general [4]. Different algorithms have been studied to build a good model to solve the problem, and in general, decision trees models have showed better results than other models, specially those used with boosting. Also, there are another good models that could be useful in our context like neural networks or linear regressions [5].

Solving the churn problem has to manage the class imbalance. A big variety of solutions have been proven as useful for solving this problem in some contexts [6]. In this case, different methods has been tested (undersampling, oversampling and one-class SVM classifiers).

Also, a big difference in the process of predicting customer churn in the Pay-TV context with respect to previous works is that the dataset does not contain any personal metric. The method presented in this paper faces all these problems by developing a decision tree with boosting and using specific Pay-TV metrics as time spent viewing Netflix, number of dispositives used by the user, tate of content viewed -considering the series and movies of the topics that the user has ever seen-, number of subscriptions levels changes and rate of content viewed entirely. The paper demonstrates that the model shows equal or better results that a combination of models using Stacking. The rest of the paper is structured as follows, first a review of works related with churn prediction in other areas are presented, then in section 3 the methodology carried out in this paper is explained and then the model defined in presented. Section 5 analysis the results and then conclusions and future works are outlined.

2. State of Art

In [7] a study about the elaboration of a model capable of predicting Customer Churn inside the telecommunication field is presented. In this study, 4 metrics groups were defined: Customer Demography -personal metrics of the customer-, Bill and Payment -payment behavior-, Call Detail Record -customer behaviour in the company services- and Customer Care Service -customer satisfaction with the company-. This model has inspired the model presented in this paper, but in the Pay-TV there is no data about the group Call Detail Record-. Therefore, instead of these information the model presented in this paper uses another type of information, called View Detail Record which includes those metrics that define the user behavior within the platform. The new category represents the same idea as the group Call Detail Record of [7], thus respecting the chosen metric structure. [8] presents a study about Customer Churn in mobile market, which uses 5 metrics groups: Demographics, Cost, Features/Marketing, Usage Level and Customer Services. By grouping the categories Cost and and Features/Marketing in a set, the result is a set of metrics very similar to the ones used in this paper.

Many studies have been done about the algorithms that can be use for predicting Customer Churn [9,10,11]. [12] presents a general summary about algorithms performance in Customer Churn prediction, and the results show that the algorithms with higher performance are Neural Networks, Decision Tree and Linear Regression. [7] predicts Customer Churn in the telecom paradigm, and it demonstrates that Decision Tree model always surpasses the Neural Network model in the prediction of churn.

Every company, to be able to perdure, needs the number of customers greater than the customers churned since otherwise the company would lose profits very quickly and would end up in bankrupt. It is because of that the Customer Churn is lower in relation with the total number of customer along the company life and that makes our dataset very imbalanced. Work with an imbalance set always causes problems [13]. Trying to solve imbalancement can cause overfitting, making the model accuracy decreases dramatically along with their capability of generalization. Class Imbalance is a very present problem in Customer Churn prediction, and many of the known techniques for solving it have been explored [5]. Among the methods proved useful it is possible to mention oversampling, undersampling and boosting, which have shown a clear improvement in the model accuracy. These are the methods tested in the model presented.

3. Methodology

The Knowledge Discovery in Databases, also known as KDD, is defined as the ”non trivial process of identifying valid, novel, potentially useful and ultimately understandable patterns of in data”. The problem faced in this work is to identify each customer as ”potentially churner” or ”potentially non churner” at the moment where the model is executed. Then, the KDD function for our problem is defined as a classification problem. Having the correct data is as important as having the correct method [14], so the first step is the adquisition and preparation of data.

3.1. Data Acquisition

In this work an actual dataset, with 300.000 customers entries along two years is used, where around 80% of the customers are non-churned and the rest are churned. The dataset belongs to a private Pay-TV company, called Mirada TV which is a leading provider of cutting-edge digital TV technology, committed to future-proofing the platforms of operators and broadcasters worldwide [1]. Experts from the company have collaborated actively in this model. Taking into account that the dataset comes from real clients, it is important to mention that very specific details of the dataset are not going to be revealed. Nevertheless, it is possible to define the groups of information used in this work, which are the following:

- **Device information:** Information about the hardware that the client is using to access the services of Mirada TV. This group of metrics can determine the economical level of the customer.
- **Bill and Payments:** Purchases and other transactions than the customer does inside the application. It can determine the satisfaction of the customer with the services, along to its economical level.
- **TV Detail Service:** How the customer uses the platform and how much he uses it. It can determine the level of satisfaction with the products of Mirada TV and the level of consumption that the customer has.
- **Errors:** Errors ocurred in any session of a customer. Errors can affect directly the customer’s view of the product. According to the Mirada TV marketing commercials, the errors may be directly related to customer churn.

Notice that, even having the fact that there are sensible information about the clients, there is not information describing the client (i. e. the age, gender of the customer, his/her economical status or laboral situation, offers from competitors, etc). This information is very significant to correctly solve this problem [7], and it represents an additional problem to deal with.

It is also important to point out that there are two different types of metrics in the dataset, which are:

[1] https://www.mirada.tv/about/

- **Variables which are dependent of the time:** There are some metrics which are dependent of the time (i.e the errors per day). Therefore in the model it is important to define a temporal window and these type of metrics would change depending on the temporal windows and their size defined.
- **Variables which are independent of time:** These metrics are totally independent of time (i.e the day the client signed up for the application) and therefore they do not depend on the temporal window defined.

3.2. Exploratory Data Analysis and Data Preparation

One of the problems that it is necessary to address in this model is caused by the very low ratio of clients that leave the platform. This problem generates a very high class imbalance problem. Section 4.2 explains this problem has been managed in this model.

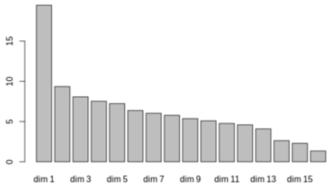


Figure 1. Variation explained for each dimension of the PCA

As explained in the previous section, due to the uses of some time dependent metrics it is necessary to define a temporal window to calculate them. The temporal window used has been defined of 6 months. Also, the dataset is reduced by eliminating the metrics that have a correlation higher than 0.9 to another metric by doing a Principal Component Analysis (PCA) (without PCA the computational cost of the model was too high because the big number of features). The result of the PCA is shown in 1. The results show that, according to the elbow method, there are two reasonable options: keep only one dimension (conserving only 20% of the variance) or keep 5 dimensions (thus conserving 51% of the variance).

4. Model

As presented in the litterature section, there are several models and techniques that have been proved useful to predict customer churn in other areas. Therefore, different alternatives -including the algorithms used and the way to process the data- have been tested to solve the problem in the area of the Pay-TV platforms. It is important to clarify that the model has been implemented using the library scikit for Python [2].

[2] https://scikit-learn.org

4.1. Techniques

In order to build the model, one classification algorithm has to be selected. In [1,7,8,12] different techniques were proved useful in similar problems. To select the technique with which to build the model, several algorithms were tested using a smaller set of the dataset, and the algorithm which yields better results has been selected.

Specificaclly next techniques were tested: neural networks (NN), K-neighbors with the variants of centroids (KNCn) and with principal componen analysis (KNCa), support vector classification (SVC) and also SVC with its linear variant (SVCL) and Nu-support vector classification (NuSVC), one class predictor (OCP), decission trees (DT) and DT with boosting based in gradients (GBC) and histograms (HGBC), and finally logistic regression (LR), and LR with crossvalidation (LRCV). All of them were tested with the defaults parameters sets by scikit-learn [15]. More details can be found in [16].

The dataset used in these experiments was a balanced set with 10.000 churned customers and another 10.000 no churned customers, selected randomly from the original dataset, therefore the dataset has been undersampled. This new dataset is only used in the scope of this section. The same partition was used for all methods, with 70% of data for training and 30% for test. Each test was executed 10 times, and the mean of the scores are shown in figure 2. The results shows that most of the algorithms exceed the 75% of score. As HGBC was the method with better results, this was the one selected to build the model.

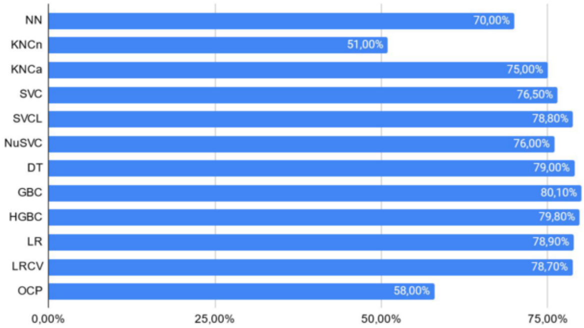


Figure 2. Results of initial test methods.

4.2. Class Imbalance Problem

As mentioned before in this paper, our original dataset is very imbalanced because the number of clients abandoning a platform is very low in comparison with the ones that remains as clients (around 80% are non-churn clients). Therefore the original dataset has a predominant class and the model will learn to predict very well this class but not the other, and for the companies the ”churn class” which is not the dominant is the most important one. There are several techniques that can apply apllied to solve this problem [13]. In this work two different aproximations has been tested: oversampling and undersampling. Both techniques have been applied to the original dataset using the HGBC algorithm for learning. In both cases the experiment was done 10 times. In average, oversampling obtains 78% score and undersampling obtain 77% score. Therefore oversampling was selected as the technique to be used in the final model of this work.

4.3. Oblivion Modeling

We humans do not remember the things that happened today with the same intensity as those that happened a week ago. Therefore, metrics that are time dependent are susceptible to be forgetable. Therefore, the model presented in this paper has tried to model the fact that people forget things by defining a ”oblivion model”. The transformation f proposed is an exponential cubic function that, given a day t_i and the value of that metric in that day $m_{j,i}$, the new value is calculated as follow

$$f(m_{j,i}) = m_{j,i} \cdot e^{\left(\frac{t_i - t_{max}}{d}\right)^3} \tag{1}$$

where d is the half of the days that the interval to take into account has and t_{max} is the value of the max day (normally, the integer value of today). By fixing the t_{max} as 1000, d as 30 and $m_{j,i} = 1$ for all i and for a given j, the resulting function showed in figure 3 is got.

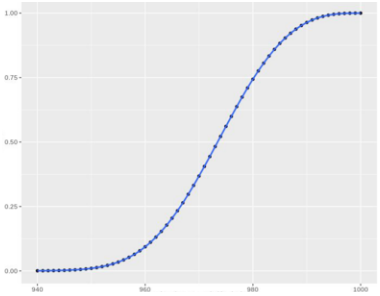


Figure 3. Function of the oblivion model.

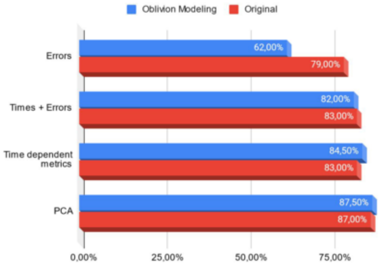


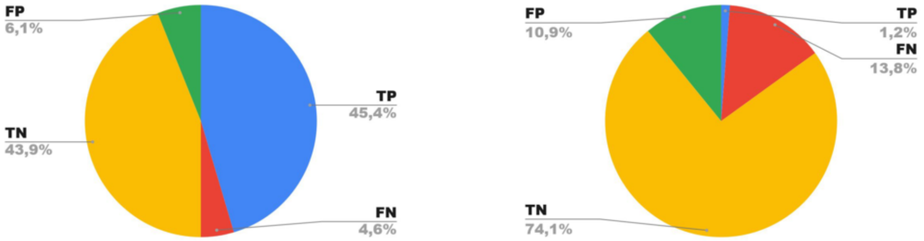
Figure 4. Comparation between the original model and with the oblivion model.

To test the utility of this model, a new test is defined. A smaller dataset created with the data of 20.000 customers is used, and partitionig it into 70% for training and 30% for test. Again 10 executions were done and saving the average scores. This dataset was used with the original model (adapting the time window to the oblivion model equivalence) and the results were compared with the new method applying transformation for several time dependent variables: first only the errors, then the errors and time spent by the customer in the platform, and finally for all the time dependent metrics. Additionally, the models were tested also applying PCA.

The results of figure 4 show that the results are better by applying this new oblivion model.

5. Results

The final model was developed using the HGBC algorithm. Their parametres were estimated using random search. The dataset used to develop the model was the dataset described in section 2, and the transformation of the oblivion model was applied (with a d of 30 days) and making a PCA with the resulting metrics. The partition defined was 60% of the dataset to train, 30% to test and 10% to validate. The model was trained 10 different times and calculating the score average. To manage the imbalance problem oversampling was used in the test and training sets, but no with the validation set in order to calculate the outperformance of the model over a real distribution of the data.



(a) On the test set

(b) On the validation set.

Figure 5. Results of the final model.

Several thresholds to decide if an instance is negative or positive have been tested, resulting that 50% worked better. The final score of the model was 86% in the test set and 90% for the validation data, as we can see in figure 5. In this model the negatives are the no churned customers and the positives are the customer that churned.

6. Conclusions

In this work, a comparative study of algorithms for predicting the customer churn in the Pay-TV sector has been done. Oversampling and undersampling methods were tested for handing the class imbalance problem inherent in this problem. A new model considering the fact that people forget things happened long time ago is presented and named the oblivion model. This model improves the results got without applying it because the use of metrics that are time dependent. Finally, model wich can discriminate the churn of customers is constructed and presented.

7. Acknowledgments

This project has been funded by the Ministry of Economy and Commerce with project contract TIN2016-88835-RET and by the Universitat Jaume I with project contract UJI-B2020-15, and by Mirada.TV.

References

[1] V Umayaparvathi and K Iyakutti. A survey on customer churn prediction in telecom industry: Datasets, methods and metrics. *International Research Journal of Engineering and Technology (IRJET)*, 3(04), 2016.

[2] Ya Gao, Guangquan Zhang, Jie Lu, and Jun Ma. A bi-level decision model for customer churn analysis. *Computational Intelligence*, 30(3):583–599, 2014.

[3] Xiaobing Yu, Shunsheng Guo, Jun Guo, and Xiaorong Huang. An extended support vector machine forecasting framework for customer churn in e-commerce. *Expert Systems with Applications*, 38(3):1425–1430, 2011.

[4] Koen W De Bock and Dirk Van den Poel. An empirical evaluation of rotation-based ensemble classifiers for customer churn prediction. *Expert Systems with Applications*, 38(10):12293–12301, 2011.

[5] Jonathan Burez and Dirk Van den Poel. Handling class imbalance in customer churn prediction. *Expert Systems with Applications*, 36(3):4626–4636, 2009.

[6] Xinjian Guo, Yilong Yin, Cailing Dong, Gongping Yang, and Guangtong Zhou. On the class imbalance problem. In *2008 Fourth international conference on natural computation*, volume 4, pages 192–201. IEEE, 2008.

[7] V Umayaparvathi and K Iyakutti. Applications of data mining techniques in telecom churn prediction. *International Journal of Computer Applications*, 42(20):5–9, 2012.

[8] Mohammed Hassouna, Ali Tarhini, Tariq Elyas, and Mohammad Saeed AbouTrab. Customer churn in mobile markets a comparison of techniques. *arXiv preprint arXiv:1607.07792*, 2016.

[9] David L García, Àngela Nebot, and Alfredo Vellido. Intelligent data analysis approaches to churn as a business problem: a survey. *Knowledge and Information Systems*, 51(3):719–774, 2017.

[10] Jaehyun Ahn, Junsik Hwang, Doyoung Kim, Hyukgeun Choi, and Shinjin Kang. A survey on churn analysis in various business domains. *IEEE Access*, 8:220816–220839, 2020.

[11] Adnan Amin, Feras Al-Obeidat, Babar Shah, May Al Tae, Changez Khan, Hamood Ur Rehman Durrani, and Sajid Anwar. Just-in-time customer churn prediction in the telecommunication sector. *The Journal of Supercomputing*, 76(6):3924–3948, 2020.

[12] John Hadden, Ashutosh Tiwari, Rajkumar Roy, and Dymitr Ruta. Computer assisted customer churn management: State-of-the-art and future trends. *Computers & Operations Research*, 34(10):2902–2917, 2007.

[13] Joffrey L Leevy, Taghi M Khoshgoftaar, Richard A Bauder, and Naeem Seliya. A survey on addressing high-class imbalance in big data. *Journal of Big Data*, 5(1):1–30, 2018.

[14] Yongbin Zhang, Ronghua Liang, Yeli Li, Yanying Zheng, and Michael Berry. Behavior-based telecommunication churn prediction with neural network approach. In *2011 International Symposium on Computer Science and Society*, pages 307–310. IEEE, 2011.

[15] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[16] Vicente López Oliva. Predicción de churn en una plataforma de pay-tv mediante machine learning. Technical report, Universitat Jaume I, 2020.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210130

On the Convergence of Financial Distress Propagation on Generic Networks

Irene UNCETA [a,1], Bernat SALBANYA [a] and Jordi NIN [a]
[a] *Universitat Ramon Llull, ESADE, Barcelona*

Abstract.
Financial networks represent the daily business interactions of customers and suppliers. Research in this domain has mainly focused on characterizing different network structures and studying dynamical processes over them. These two aspects, structure and dynamics, play a key role in understanding how emergent collective behaviors, such as those that arise during economic crises, propagate through networks. Business interactions between companies form a direct and weighted network, where the financial distress of a node depends on the ability of its customers to fulfill payments. In situations where there is no such inbound cash flow, a company may have to close down due to a lack of liquidity. Interconnection therefore seems to be at the core of systemic fragility. Whether the nature and form of this connection may have an impact on how distress is propagated is still an open question. In this paper, we study how disruptive events propagate through different network structures, under different scenarios. For this purpose, we use a liquidity model that describes how the economy of nodes evolves from a given initial state in terms of their interactions. From our experiments, we empirically conclude that most of the studied network dynamics reach a steady-state, even in the presence of large noise values.

Keywords. Non-linear systems, financial networks, convergence testing

1. Introduction

In a customer-supplier network, nodes refer to individual firms and connections map their interactions or economic exchanges. This representation has allowed researchers to study how information, money, materials or components can flow through networks of different shapes and structures [1]. In the case of financial markets, a network approach has been particularly successful in characterizing the dynamics of customer-supply networks. These networks play a fundamental role in the optimization of production chains and the analysis of systemic risk [2]. Hence, studying the dynamics of these networks is of outermost importance to predict emergent phenomena that could potentially have negative consequences on individual nodes, as well as on the economy as a whole [3]. Even small disturbances can drive supply networks away from their desired state and towards unstable regimes [4] where disruptions are amplified through the network [5].

[1] Corresponding Author: Irene Unceta, Department of Operations, Innovation and Data Science, Universitat Ramon Llull, ESADE, Barcelona; E-mail: irene.unceta@esade.edu.

Financial distress propagation has been studied using different models and network structures [6]. Recent studies have shown that there exist significant correlations between the local topological properties of nodes in a financial network and their risk of default [7]. Other studies have focused on understanding how this risk is propagated through the network, leading to scenarios of systemic instability. Or, even, on understanding how a node's economy state evolves in time.

In particular, in [8] authors introduced an additive economic model that described how node liquidity evolves in time using an equation that captures both exchanges between individuals and random speculative trading. This model has been assayed on a synthetic, heterogeneous (power-law) network. In this type of network, some nodes are highly connected, i.e. they have many links to other nodes. Yet, the overall number of connections is low. As a result, phenomena such as preferential attachment emerge, whereby new nodes tend to connect to nodes with larger degrees, i.e. hubs. In line with these property, authors observed that a few agents tended to concentrate the majority of the wealth, following a Pareto-tail distribution. This conclusion was drawn by using a simple but effective equation where trading was introduced as two independent additive noises over time. This model has been then further refined and adapted to study distress propagation in [9]. In this follow-up work, the authors have shown that the proposed model reaches a steady-state even in the presence of large noise volatility values, when the system is composed of a single agent and a market node.

In this paper, we build on the intuition introduced in these two papers to evaluate the convergence of the described model for different network architectures. To this end, we expose a synthetic network, with initially given liquidity conditions, to specific disruption events, and simulate the evolution of the liquidity distribution until its convergence. Notably, we study the evolution of the liquidity distribution for different generic network topologies, in different scenarios that may or may not include a market node. From our experiments, it is possible to extract meaningful insights about how the topology affects distress propagation and what the impact of a market node is for ensuring resilience against disruption.

2. Materials and methods

Time evolution of a financial network can be monitored by modelling the interactions between the nodes at each time step and updating each node's state accordingly. The state of a node can be related to any given property. For example, its liquidity.

2.1. Liquidity model

Following [8], we can define the *liquidity* $L(t)$ of a node as its amount of disposable money. The liquidity of a node has a direct relationship with two different elements: the economic exchanges between the considered node and the other nodes in the network, and the market volatility. The economic exchanges are usually represented by the edges of the considered network. The volatility, in turn, accounts for the uncertainty of the market, and is defined as an external factor affecting each node independently. We can approximate it by means of a Gaussian distribution.

To simulate the evolution of the liquidity over time for individual nodes, we introduce a *liquidity model* that describes how the economic activity of each node evolves in terms of its interactions with other nodes, given an initial state. This initial state $L(0)$ stands for the initial liquidity conditions of each node at time 0. The larger its value, the more money the considered node will dispose of to face potential economic volatility, crises, and changes in the money flows through the network. Given this initial state, we can compute the liquidity $L_i(t)$ of a node i at any given point in time t as:

$$L_i(t+1) = L_i(t)\,(1+\eta(0,\sigma)) + \sum_{j\in N_i} w_{ji}P(t)H(L_j(t)-w_{ji}) - \sum_{j\in N_i} w_{ij}P(t)H(L_i(t)-w_{ij})$$

where $\eta(0,\sigma)$ stands for the economic volatility represented as a Gaussian noise with mean 0 and standard deviation σ, N_i stands for the set of neighbors of node i, w_{ji} and w_{ik} correspond to the weights of an edge from j to i and vice versa, $P(t)$ is a random variable which follows a Bernoulli distribution with probability p and, finally, $H(L_i(t)-w_{ij})$ represents a Heaviside step function which evaluates to 1 if the node i has enough liquidity to afford the considered money transaction and 0 otherwise.

A Gaussian distribution governs market volatility for each node. Following the parameters of this distribution, a random normal value is given to every single node for each time-step simulation. Its impact on the liquidity is proportional to the liquidity at the previous time step and may affect the outcome positively or negatively. For this reason, the mean of the Gaussian distribution is set to zero ($\mu = 0$).

The money exchanges are, in turn, controlled by the global parameter p, corresponding to the probability of a Bernoulli distribution. The value of p ranges from 0 to 1. When $p = 1$, the network is fully activated, meaning that all money exchanges take place. For lower values of p, the network is not fully activated, i.e., certain due payments never take place. Therefore, p can be understood as the fraction of active exchanges at any time.

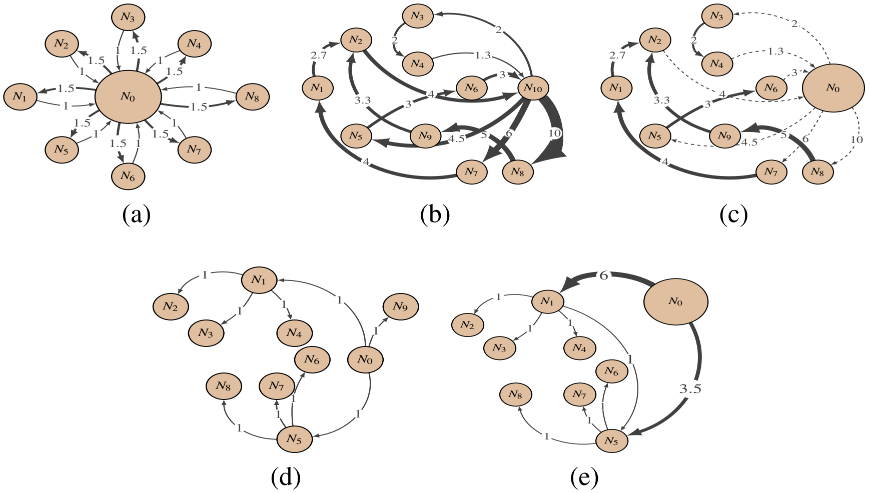
The equation above allows us to simulate the evolution of a given network in time under different initial conditions, encoded by the initial state $L_i(0)$ for each node. Depending on the considered network structure, we can study the effect of these conditions in the overall state of the network by the time we reach convergence. In [9], authors have shown that a single node network following this model reaches a steady-state even in the presence of large noise volatility values. For that network, the interplay between the outbalance of the initial income and outcome money flows, and the multiplicative noise level σ leads to a symmetry in the final liquidity distribution. Here, we extend this single node model system to more complex scenarios and assay this model in under different economic conditions. These conditions are defined by different initial states and activation levels of the network. For each set of conditions we simulate the evolution of liquidity in time, and verify that the proposed model converges to a stationary state.

2.2. Scenarios

In some cases we introduce a *market node*. This node accounts for unknown transactions between the nodes of the network, i.e. for economic transactions conducted elsewhere. Including these data is important to obtain a reliable approximation of the economic state of each node in time. Combined with a large activation of the system, which means nodes are capable of doing exchanges, a market node decreases the overall distress probability of the considered systems.

In our experiments we have assumed that the market node has infinite liquidity and that it transfers money to the other nodes with a liquidity ratio equal to 1.5. In practice, this means that for a full activation of the network ($p = 1$), individual nodes earn 1.5 from their connections, including the market node, and pay back 1. Therefore, in each time-step, profits are set to 0.5. Considering the amount of taxes and benefits that each node must pay and earn, we consider this to be a realistic value [9]. Moreover, this allows us to simulate economic exchanges within a growing economic context and avoids a general default of the system, which is seldom observed in practice.

The first scenario we consider is a *Star network*. This network is an extension of the single node network, where the different nodes are connected forming a star, as shown in Figure. 1 (a). In this structure, each N_i node is linked to the market node N_0, and is independent of the remaining nodes N_j for $j \neq i$, meaning that there is no exchange of money among them. The market node provides liquidity to nodes and receives the payments from it.



(a) (b) (c)

(d) (e)

Figure 1. Topology of the simulated networks: (a) Star, (b) Erdos-Renyi without market node, (c) Erdos-Renyi with market node, (d) Power Law without market node, (e) Power Law with market node.

A more complex monetary exchange is that of an *Erdos-Renyi network*, where there exist random edges among the nodes. In this configuration, any node N_i is linked with the same probability to the other nodes. Each edge is included in the network with a certain probability, which is independent for each edge. Therefore, each node is statistically independent of the rest of N_j nodes for $j \neq i$. This structure represents a collaborative economy, where money flows through the agents. In this paper we study this network first in absence of a market node, as shown in Figure. 1 (b). We later introduce this market node as displayed in Figure. 1 (c) and evaluate its impact on the overall dynamics. When having the market node N_0, we assume all the remaining nodes are linked to it.

Finally, a more realistic scenario is that of a *Power Law network*. This type of network replicates an economic structure that includes both prevalent and secondary companies, as generally observed in real markets. Some nodes N_i are highly connected, meaning they have many links to other N_j nodes for $j \neq i$. Yet, the overall number of connections among all nodes is low. We can understand this configuration as representing a capitalist economy. Similarly to the Erdos-Renyi case, we consider this network with and without a market node. In the former case, all nodes are connected to the market node N_0 to keep a constant liquidity ratio.

We have created synthetic networks according to the five scenarios described above. In what follows, we study the convergence of each system's liquidity and analyse how disruptive events propagate through the different network topologies.

3. Numerical Simulations on Generic Networks

For the case of the Star network, we have created simulated scenarios with 100 nodes. For both the Erdos-Renyi and the Power-law configurations, we have included up to 1,000 nodes. For the latter case, each new node adds three random edges, with a probability of adding a new triangle after adding an edge of 30%.

For each of the five configurations, we have conducted 1,000 simulations during 5,000 time-steps for a volatility $\sigma = 0.05$ and $\sigma = 0.3$. These values are chosen following [9] in order to represent two scenarios: low and high volatility. Or in other words, stable and unstable economies. We have assumed every single node to have enough liquidity to withdraw the first installment as an initial state. During each time step, we have simulated money exchanges according to the Bernoulli probabilities associated with the different nodes and computed their liquidity in time. Throughout the simulations, we have allowed nodes to be indebted and to recover from this state.

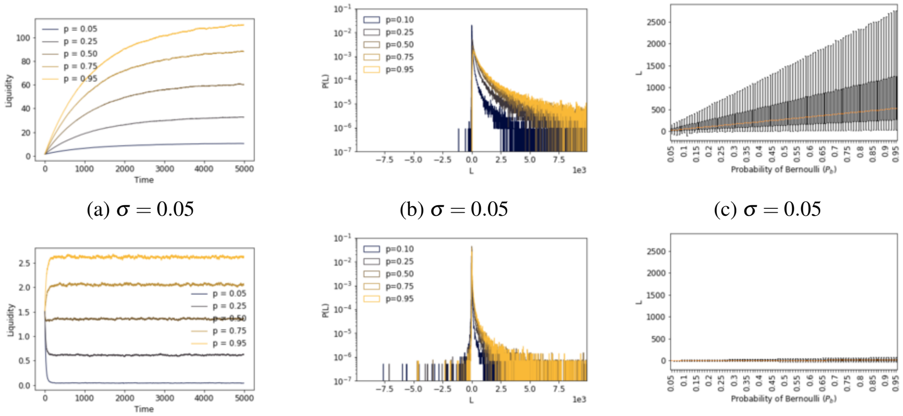
3.1. Discussion of results

First, we evaluate the convergence of the median node liquidity in time for different values of the Bernoulli probability p. In most cases, we observe a steady-state around 5,000 time-steps, when the median of the liquidity stabilizes. We also study the impact of disruptions for the networks described in the previous section by analyzing the liquidity distribution at the steady-state for each Bernoulli probability p, across 1,000 simulations.

3.1.1. Star Network

In order to analyze the effect of a central market node ruling the economy, we now study the evolution of the liquidity for the Star network scenario. For both $\sigma = 0.05$ and $\sigma = 0.3$, we observe that the median of the liquidity tends to a steady state, as shown in Figure. 2 (a) and (d), for the different values of the Bernoulli probability p. Depending on the value of p, we observe different growth rates of the median liquidity. For this network, the market node supplies liquidity every time-step to the individual nodes. However, this growth rate depends on the activation of the system. Thus, for higher activation values the median liquidity is greater than for low activations.

As observed for the previous case, the liquidity reaches a steady state for any value of p. The speed with which this steady state is reached depends on the volatility. For $\sigma = 0.05$, the system reaches the steady state around $t = 5,000$, whereas for $\sigma = 0.3$ this happens around $t = 200$. Moreover, for this case, we observe a decrease for lower activation rates. This is due to the greater market shocks that, combined with fewer transactions, make nodes more susceptible to falling into indebtedness.



(a) $\sigma = 0.05$ (b) $\sigma = 0.05$ (c) $\sigma = 0.05$

(d) $\sigma = 0.3$ (e) $\sigma = 0.3$ (f) $\sigma = 0.3$

Figure 2. (a)(d) Median of the total liquidity at time t for different combinations of Bernoulli probability p for a Star network with market. (b)(e) Distribution of the system liquidity at time $T = 5,000$. Note that, y-scale is logarithmic. (e)(f) Distribution of the system liquidity at time $T = 5,000$.

Figures. 2 (b) and (e) show the distributions of the core system liquidity at the steady state. The liquidity distributions are not symmetric for any values of σ and p, although we still observe a peak at $t = 5,000$. The peak increases and the tails flatten as volatility increases, having more portion of nodes with debts in this case.

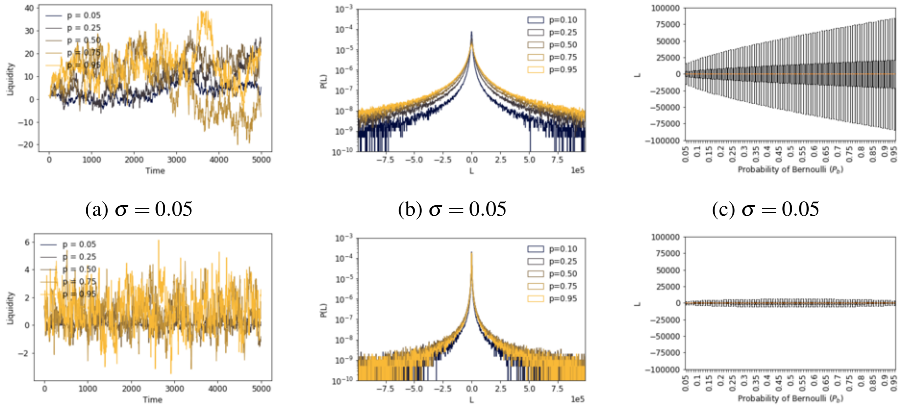
In Figure. 2 (c) and (f), where the boxplots display the different liquidity values at the steady state across the different values of p, we also observe the asymmetry in the liquidity distribution and the growth in the liquidity median. The injection of credit by the market node and a major activation shape the liquidity distribution. A smaller spread for high volatility is a consequence of greater market shocks.

3.1.2. Erdos-Renyi Network

We can represent a collaborative economy by means of an Erdos-Renyi network. In the absence of a market node, the behavior of the liquidity is completely random, and it does not converge to a steady state, as we can observe in Figures. 3 (a) and (d), where the median liquidity values are plotted in time for volatility $\sigma = 0.05$ and $\sigma = 0.3$. These show that an Erdos-Renyi without a market acts as a random and isolated system.

In Figures. 3.(b) and (e), the distributions of the core system liquidity at the steady state are displayed. We observe a symmetry in the liquidity distribution at time $t = 5,000$. That is, for higher volatility, the peak of the distribution increases, whereas the tails have a lower spread. Moreover, some differences are observed in the liquidity boxplots for different p at the steady state depicted in Figures. 3.(c) and (f). Here, for $\sigma = 0.05$, the maximum spread of the liquidity is for the greatest activation of the system ($p = 1$). Yet for $\sigma = 0.3$, the maximum swifts to $p = 0.5$.

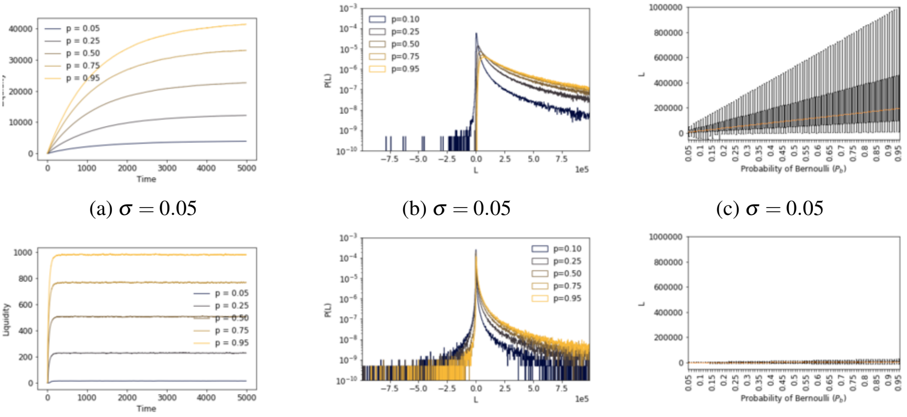
Significant differences are observed when a market node is added to the Erdos-Renyi network. Figures. 2 (a) and (d) show that the median of the liquidity tends to a steady state for both $\sigma = 0.05$ and $\sigma = 0.3$ and the different values of the Bernoulli probability p. Again, the presence of a market node has an effect on the evolution of liquidity, with different growth rates for the different values of p, as observed for the Star network.



(a) $\sigma = 0.05$ (b) $\sigma = 0.05$ (c) $\sigma = 0.05$

(d) $\sigma = 0.3$ (e) $\sigma = 0.3$ (f) $\sigma = 0.3$

Figure 3. (a)(d) Median of the total liquidity at time t for different combinations of Bernoulli probability p for an Erdos-Renyi network without market. (b)(e) Distribution of the system liquidity at time $T = 5,000$. Note that, y-scale is logarithmic. (e)(f) Distribution of the system liquidity at time $T = 5,000$.



(a) $\sigma = 0.05$ (b) $\sigma = 0.05$ (c) $\sigma = 0.05$

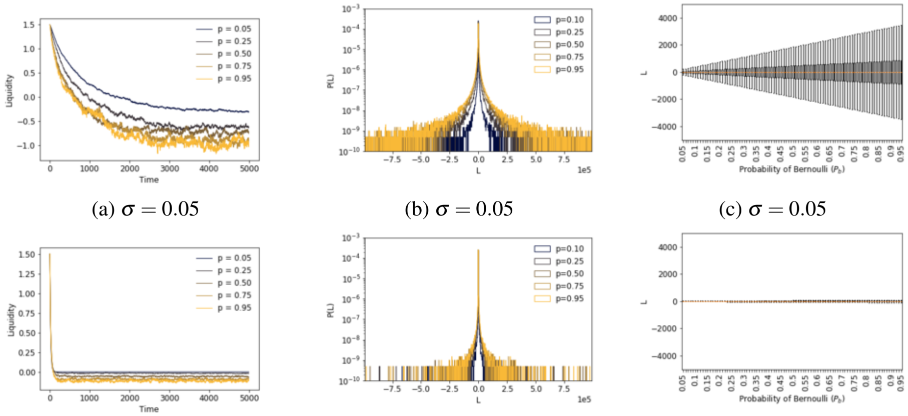
(d) $\sigma = 0.3$ (e) $\sigma = 0.3$ (f) $\sigma = 0.3$

Figure 4. (a)(d) Median of the total liquidity at time t for different combinations of Bernoulli probability p for an Erdos-Renyi network with market. (b)(e) Distribution of the system liquidity at time $T = 5,000$. Note that, y-scale is logarithmic. (e)(f) Distribution of the system liquidity at time $T = 5,000$.

Similarly, the system reaches the steady state around $t = 5,000$ for $\sigma = 0.05$, whereas only 200 time-steps are needed for $\sigma = 0.3$. Contrarily, we do not observe a decrease for lower activation rates in these cases. The inter-connectivity of the different individual nodes, makes this network more resilient to distress propagation.

As opposed to the cases discussed above, where the distribution of the overall liquidity value was symmetric around 0, for the case of the Erdos-Renyi network with a market node, shown in Figures. 4.(b) and (e), we observe a right-tailed liquidity distribution. This asymmetry is greater for larger values of the Bernoulli probability p. However, differences observed for the different values of p become smaller as the volatility increases. Both right and left tails flatten with increasing values of σ, whereas the peak increases, as was seen before. This asymmetry can also be observed in Figure. 4.(c) and (f), where boxplots show the distribution of liquidity for different activation probabilities.

3.1.3. Power Law Network



(a) $\sigma = 0.05$ (b) $\sigma = 0.05$ (c) $\sigma = 0.05$

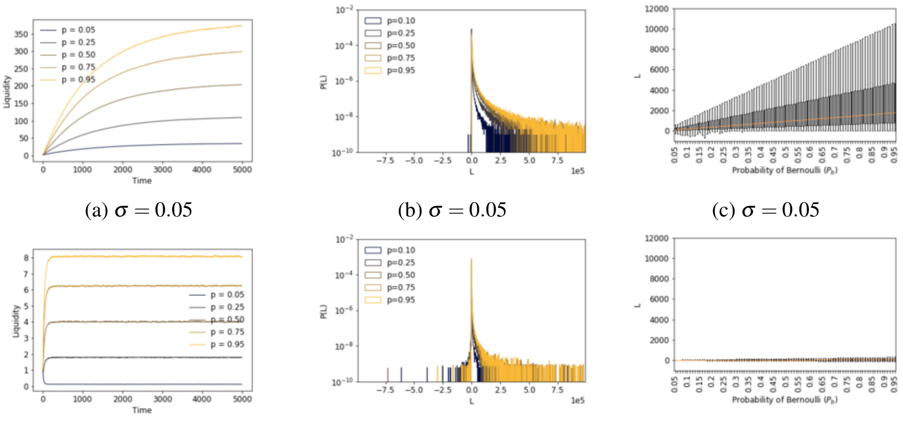
(d) $\sigma = 0.3$ (e) $\sigma = 0.3$ (f) $\sigma = 0.3$

Figure 5. (a)(d) Median of the total liquidity at time t for different combinations of Bernoulli probability p for a Power Law network without market. (b)(e) Distribution of the system liquidity at time $T = 1000$. Note that, y-scale is logarithmic. (e)(f) Distribution of the system liquidity at time $T = 1000$.

Finally, we represent a capitalist economy using a Power Law network. For such network without the market node, we observe a decrease to a steady state for any value of p. Figure. 5 (a) and (d)) show the evolution of the median of the total liquidity in time for the different values of the Bernoulli probability p and for different volatility values. We observe a faster decrease in the liquidity median over time as the higher is the volatility. Thus, for $\sigma = 0.05$, the system reaches the steady state around $t = 4,000$, whereas for $\sigma = 0.3$ this happens around $t = 200$. The distribution of liquidity at the steady state for different p is still symmetric, as depicted in Figure. 5.(b) and (e). Moreover, we observe a similar behavior of the distribution of the liquidity as in the one node setting: for higher volatility, the peak of the distribution increases (around 1.5 in this case), whereas the tails have a bigger spread.

These new liquidity distributions are different because of the combination of the number of nodes and volatility. With 1,000 nodes, the aggregated Bernoulli probability of all nodes acts as a Binomial distribution, which occurs for both the earnings and payments. The spread of a Binomial distribution increases with p, as it can be observed in Figure. 5(c) and (f), which display the boxplots of the liquidity distribution across that variable. We can confirm this for $\sigma = 0.05$ when the effect of volatility is very small. The spread of the liquidity is maximal for $p = 1$, whereas for $p = 0$ is minimal. For higher volatility, the multiplicative effect of σ shapes the distribution of the liquidity, with a maximum in $p = 0.5$. Therefore, we can conclude that the number of active edges matters concerning the range of the possible liquidity values.

In a Power Law network with 1,000 nodes, we have fewer edges than in an Erdos-Renyi network with the same amount of nodes. The effect of this can be observed in Figure. 5(c) and (f), where the absolute amounts of liquidity are much lower than those observed for the Erdos-Renyi. Thus, the effect of the multiplicative noise (σ) is also less significant, being the spread of the liquidity driven by the Binomial distribution. For this system, we have a liquidity ratio greater than 1, but no liquidity from the market. This network acts at the end as an isolated system. The symmetry of the liquidity distribution is still present for any volatility.



(a) $\sigma = 0.05$

(b) $\sigma = 0.05$

(c) $\sigma = 0.05$

(d) $\sigma = 0.3$

(e) $\sigma = 0.3$

(f) $\sigma = 0.3$

Figure 6. (a)(d) Median of the total liquidity at time t for different combinations of Bernoulli probability p for a Power Law network with market. (b)(e) Distribution of the system liquidity at time $T = 1000$. Note that, y-scale is logarithmic. (e)(f) Distribution of the system liquidity at time $T = 1000$.

When analyzing the results for the case where we include the market node, we again observe a growth for the liquidity median of all nodes. The evolution of the median of the liquidity of all nodes is shown in Figure. 6.(a) and (d), where we observe a faster increase in the liquidity median over time as higher is the volatility. As for a Power Law without a market node, for $\sigma = 0.05$, the system reaches the steady state around $t = 4,000$, whereas for $\sigma = 0.3$ this happens around $t = 200$. Figure. 6.(b) and (e) depicts an asymmetry of the distribution of liquidity at the steady state for different values of p caused by the presence of the market node. The spreads are smaller due to the fewer number of edges between nodes in the Power Law network. Both right and left tails flatten as higher is the σ, whereas the peak increases, as it was seen before. Despite the effect of the market node, the spread of the liquidity for a Power Law is still driven by the Binomial distribution, and volatility has a smaller consequence compared to the Erdos-Renyi network. In Figure. 6.(c) and (f), the alternative representation of the distribution of liquidity at the steady state with boxplots confirms the asymmetry for any volatility.

4. Conclusions

In this article, we have studied how the risk of default propagates in different synthetic financial networks by using a liquidity model. Networks are a valid representation of economic transactions such as wires to pay for services or goods. Based on previous works on wealth distribution, the used model combines two stochastic terms: an additive noise accounting for the capability of trading and paying obligations, regulated by a parameter

p, and a multiplicative noise representing the variations of the market, controlled by a parameter σ.

Our results show that, even in the presence of large noise values, network dynamics reach a steady-state, with the only exception of the Erdos-Renyi network without market node. In this sense, the presence of a market node seems crucial for the stability of the system. The fact that a market node injects money to nodes constantly causes an increase in the median liquidity. In contrast, when there isn't such a market node, the total system liquidity remains constant for any p. For this reason, we do not observe significant differences between the behavior for low and high Bernoulli probability (p) where there is a network without the market. Therefore, in that case, the liquidity ratio and the volatility (σ) control the amount of money the nodes can accumulate. In spite of the median liquidity distribution being comparable for all the networks whenever a market node is included, differences arise in the absence of this node. This is, we detect a network effect when there is no market node. When this happens, the different connections between nodes due to the different network topologies have a bigger impact. We conclude that inter-connectivity is crucial for the time evolution of liquidity.

From an economic perspective, the network structure of economic interactions and its combination with the system volatility (σ) is crucial to determine the maximum default probability. We conclude that this model can be a valuable tool to study systemic risks of different economic systems. In this sense, Artificial Intelligence may help us to understand the behavior of Financial Networks. Future research following this work should focus on studying strategies to prevent risk of default and cascade effects in the different settings.

In addition, further research should also study the effect of an optimal rewiring of nodes from an agent-based perspective, to design protocols that ensure resilience against disruption propagation.

References

[1] E. Atalay, A. Hortasu, J. Roberts, and C. Syverson. Network structure of production. *Proceedings of the National Academy of Sciences (PNAS)*, 108(12):5199–5202, 2011.

[2] A. Barja, A. Martínez, A. Arenas, P. Fleurquin, J. Nin, J. Ramasco, and E. Tomás. Assessing the risk of default propagation in interconnected sectoral financial networks. *EPJ Data Science*, 8(1):32, 2019.

[3] F. Corsi, F. Lillo, D. Pirino, and L. Trapine. Measuring the propagation of financial distress with granger-causality tail risk networks. *Journal of Financial Stability*, 38:18–36, 2018.

[4] C. Bode and S.M. Wagner. Structural drivers of upstream supply chain complexity and the frequency of supply chain disruptions. *Journal of Operations Management*, 36:215–228, 2015.

[5] K Zhao, Z Zuo, and JV Blackhurst. Modelling supply chain adaptation for disruptions: An empirically grounded complex adaptive systems approach. *J. of Operations Management*, 65(2):190 212, 2019.

[6] L. Veraart. Distress and default contagion in financial networks. *Math. Finance*, 30:705–737, 2020.

[7] E. Letizia and F. Lillo. Corporate payments networks and credit risk rating. *EPJ Data Sci.*, 8(21), 2019.

[8] J. Bouchaud and M. Mézard. Wealth condensation in a simple model of economy. *Phisica A*, 282:536545, 2000.

[9] J. Nin, B. Salbanya, P. Fleurquin, E. Tomas, A. Arenas, and J. Ramasco. Modelling financial distress propagation on customer-supplier networks. *Chaos Journal*, 31(5), 2021.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210131

Supporting Enrollment in Higher Education Through a Visual Recommendation System

Julià MINGUILLÓN [a,1], Noe RIVAS [a] and Jonathan CHACÓN [b]
[a] *Universitat Oberta de Catalunya, Spain*
[b] *Elisava, Spain*

Abstract. Nowadays, most universities offer programmes and subjects online, specially in the case of fully online open/distance universities. Students have a higher degree of flexibility, which allows them to choose among an endless list of subjects for advancing within their degree. Although this can be seen as a positive result of enrollment flexibility policies, it may be also the source of one of the most well-known problems in open/distance education: high dropout rates, partly caused by inadequate enrollment. In this paper we propose a recommendation system that helps students to navigate through the list of available subjects using a visual metaphor, taking into account students' preferences and previous enrollment data. Our system is based on a two-dimensional map (2D) where subjects that can be taken together appear close to each other, as neighboring regions.

Keywords. recommendation system, visualization, enrollment, higher education

1. Introduction

Traditionally, distance and open universities provide students with greater flexibility during enrollment than brick-and-mortar universities. Students have almost no requirements about the number or kind of subjects they can take during one academic semester, although different recommendations are provided, in form of static information (from official degree documentation in institutional web pages) and dynamic information (from official mentors or other peers). Nevertheless, this flexibility can be misunderstood by some students, leading them to take wrong decisions regarding enrollment (too many subjects or inappropriate combinations of subjects). Previous work in this topic [1] showed that most students make decisions about enrollment taking into account their available time, semester organization and subject characteristics. In this paper we propose a recommendation system that helps students to navigate through the list of available subjects using a visual metaphor, namely a map. Subjects become regions in a 2D map, organized according to two complementary premises: subjects that are not supposed to be simultaneously taken should appear far from each other (i.e. they should not be neighboring regions), while subjects that are safe to be simultaneously taken may appear close to each other. In order to do so, we propose to combine different criteria related to user needs in a

[1] Corresponding Author: Universitat Oberta de Catalunya, Rambla Poblenou 156, 08018 Barcelona, Spain; E-mail: jminguillona@uoc.edu

subjects' distance matrix. According to their preferences, students can specify the partial weight of each distance criterion (based on historical data, including subject difficulty, satisfaction, semestral organization, etc. [1]) and visually determine the most appropriate subjects to enroll according to their previous achievements and goals.

2. Recommendation systems in educational scenarios

In a recent review on academic advising systems [2] the authors found that recommending subjects is a primary research objective. Recommendation systems are classified in five different categories, according to their nature: Content-based, Collaborative filtering-based, Knowledge-based, Hybrid and Computational intelligence-based. Unlike other popular recommendation systems used in Amazon, Spotify or Netflix, providing students with appropriate subject recommendations needs to take into account not only students' preferences and background but also other constraints related to university policies and semestral organization, among others, that is, contextual information becomes important to provide users with good recommendations [3]. Actually, most recommendation systems can be considered hybrid in practice, as they combine aspects from the different categories. In the case of enrollment:

- Content-based: some subjects are part of the same learning path, develop the same competencies or can be prerequisites for other subjects.
- Collaborative-based: previous enrollment data and academic performance of students with similar enrollment patterns.
- Context-based: for instance, some subjects are offered only in one semester.

In [4] the authors showed that learning dashboards and the use of data visualization in educational scenarios is still limited. More recently, in [5] the authors describe a system that takes historical academic data and allows students to select subjects and predict their performance, using classical tables and line charts as a visualization. Instead, we would like to represent "time" (i.e. sequence of subjects) in a more organic way, making the student understand the concept only by taking a quick look. In order to do so, we propose a different approach, using a visual metaphor (namely, a map) that helps students to decide which subjects they want to enroll into. In summary, subjects that can be taken at the same time should appear close to each other, while subjects that should not be taken at the same time should appear far from each other. Unlike [5], we do not want to force absolute positions of subjects, but make them be part of a 2D map where each subject becomes a region, surrounded by neighboring subjects.

3. System design

Students should be able to see a personalized 2D map according to their interests and previous enrollment data and academic results. As we pursue interpretability over accuracy [6], we focus on "typical" students and "reasonable" enrollment patterns only. In order to do so, we follow Shneiderman's mantra: Overview first → Zoom and filter → Details on demand, by means of an interactive learning dashboard [4].

3.1. Data

We used enrollment data from a Computer Engineering degree extracted from the institutional learning repository store [7]. Between the academic years 2010-2011 and 2019-2020, a total of 10,957 students enrolled 42,889 times, generating 10,409 different enrollment patterns, choosing among 695 different subjects (from all the programmes offered by the university). Actually, 7,418 enrollment patterns were taken only once, showing the underlying long tail distribution. As we want to build an usable recommendation system, we discarded data from those students enrolling into 7 or more subjects (less than 1.1% of students), and the long tail of subjects with less than 100 enrollments, so only the 54 most popular subjects were taken into account.

3.2. Distance criteria

The following criteria C (described in [1]) were used to compute the 54×54 partial distance matrices D_C between each pair of subjects i and j:

1. D_S: Semestral organization ($S_{i,j}$), subjects that are not supposed to be taken together according to the degree semestral organization should appear far from each other (i.e. separated).
2. D_P: Popularity ($P_{i,j}$), subjects that are not usually taken together by students should appear separated. This factor is opposed to the previous one.
3. D_D: Difficulty ($D_{i,j}$), subjects most likely to fail when taken together should appear separated.
4. D_R: Requisites ($R_{i,j}$), subjects that need to be taken in a particular order (i.e. prerequisites) should appear separated.
5. D_O: Overlap ($O_{i,j}$), subjects that have a high overlap between their assessment activity calendars should appear separated.

The resulting distance matrix D is as a linear combination of partial distance matrices D_C, where $w_C \in [0,1]$ is the weight assigned by the student to criterion C from the previous list and ε is a small random amount used to avoid zero distances that would generate subject overlaps:

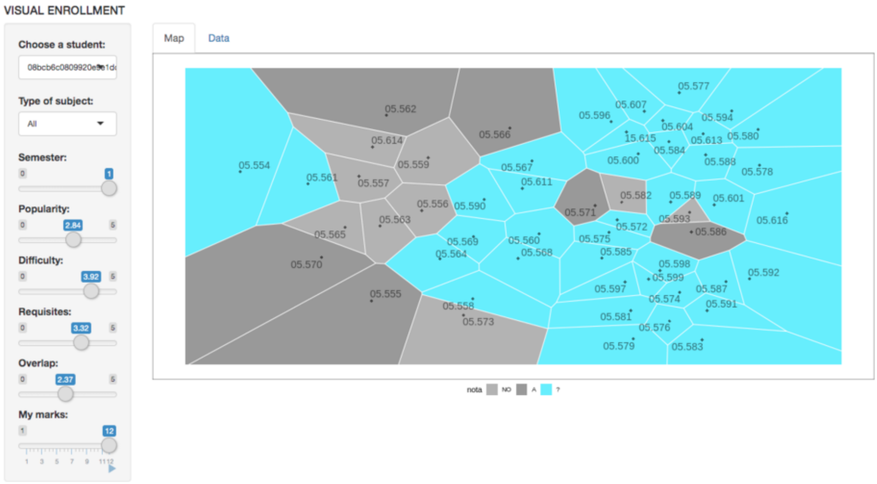
$$D = \sum_C w_C \, D_C + \varepsilon$$

Then, using Sammon's non-metric multidimensional scaling algorithm [8] we obtain points in a 2D space, which are rotated to ensure that the degree capstone project is in the right part of the map. Finally, a Voronoi diagram is computed as the final 2D map.

3.3. Prototype

Currently now, an interactive proof-of-concept developed in R and Shiny is available[2] to demonstrate the possibilities of this approach, as shown in Figure 1. The map shows student's achievements (in shades of gray) and the rest of subjects are shown in blue, according to the weights given by the student to each criterion C. Subjects in blue closest to subjects in gray are candidates for recommendation and can be easily identified.

[2] http://personal.uoc.edu:8080/VE/



VISUAL ENROLLMENT

Choose a student: 08bcb6c0809920e9e1d

Type of subject: All

Semester: 0 – 1

Popularity: 0 – 2.84 – 5

Difficulty: 0 – 3.92 – 5

Requisites: 0 – 3.32 – 5

Overlap: 0 – 2.37 – 5

My marks: 1 – 12

Figure 1. Current prototype showing data from a real student.

4. Conclusions

As an ongoing research project, several questions still need to be formally posed. For instance, how can we evaluate the proposed recommendation system? By means of student's satisfaction when using the tool? By comparing to student's real enrollment data after using our system? By measuring academic performance at the end of the semester? On the other hand, shall we use all available enrollment data (10 years, i.e. 20 semesters) or only data from the last N semesters? Is there an optimal w_C configuration?

References

[1] Rivas N, Minguillón J, Chacón J. ENROLLING HABITS IN HIGHER EDUCATION. WHAT SOURCES OF INFORMATION DO STUDENTS HAVE AND WHAT ARE MISSING? In: INTED2021 Proceedings. 15th Int. Technology, Education and Development Conf. IATED; 2021. p. 4980–4988.

[2] Iatrellis O, Kameas A, Fitsilis P. Academic advising systems: A systematic literature review of empirical evidence. Education Sciences. 2017;7(4):90.

[3] Kulkarni S, Rodd SF. Context Aware Recommendation Systems: A review of the state of the art techniques. Computer Science Review. 2020 Aug;37:100255.

[4] Schwendimann BA, Rodriguez-Triana MJ, Vozniuk A, Prieto LP, Boroujeni MS, Holzer A, et al. Perceiving learning at a glance: A systematic literature review of learning dashboard research. IEEE Transactions on Learning Technologies. 2016;10(1):30–41.

[5] Castells J, Mohammad PD, Galárraga L, Méndez G, Ortiz-Rojas M, Jiménez A. A Student-oriented Tool to Support Course Selection in Academic Counseling Sessions. In: Proceedings of the Workshop on Adoption, Adaptation and Pilots of Learning Analytics in Under-represented Regions (co-located with the 15th European Conference on Technology Enhanced Learning 2020); 2020. p. 48–57.

[6] McNee SM, Riedl J, Konstan JA. Being accurate is not enough: how accuracy metrics have hurt recommender systems. In: CHI 06 Extended Abstracts on Human Factors in Computing Systems. CHI EA 06. Association for Computing Machinery; 2006. p. 10971101.

[7] Minguillón J, Conesa J, Rodríguez ME, Santanach F. 8. In: Spector JM, Kumar V, Essa A, Huang YM, Koper R, Tortorella RAW, et al., editors. Learning Analytics in Practice: Providing E-Learning Researchers and Practitioners with Activity Data. Springer; 2018. p. 145167.

[8] Sammon JW. A Nonlinear Mapping for Data Structure Analysis. IEEE Transactions on Computers. 1969 May;C18(5):401409.

M. Villaret et al. (Eds.)

doi:10.3233/FAIA210132

Data-Driven Analysis of Friction Stir Welding for Aerospace Applications

Marta CAMPS[a,1], Maddi ETXEGARAI[a], Francesc BONADA[a], William LACHENY[b], Dorick BALLAT-DURAND[b], Sylvain PAULEAU[b] and Xavier DOMINGO[a]
[a] *Eurecat, Centre Tecnològic de Catalunya, Unit of Applied Artificial Intelligence , Av. Carrer de Bilbao, 72, 08005 Barcelona, Spain*
[b] *Ariane Group, 51/61 route de Verneuil, Bâtiment 71 - Bureau 142, 78131 – Les Mureaux – France*

Abstract. Industry 4.0 and the digitalization of the manufacturing processes have brought new opportunities and strategies for process control and optimization. Friction Stir Welding is becoming a relevant manufacturing technology for several applications, among them the aerospace sector. This work presents the first data analysis and characterization of the Friction Stir Welding process of the Pre-Final Assembly Line of the new Ariane 6 launcher. Process monitoring data is captured and analyzed to provide predictive quality solutions for improving manufacturing key performance indicators and bring smart manufacturing and Industry 4.0 digitalization into the aerospace manufacturing sector. The results show promising performance for abnormal behavior detection, leveraging on a tailored data manipulation approach for this unique use case.

Keywords. Feature Engineering, Machine Learning, Predictive Quality, Industry 4.0

1. Introduction

The proliferation of digitalization in manufacturing processes, fostered by Industry 4.0, has brought a new set of functionalities [1] with the potential to transform production by reinventing the monitorization, control and optimization of the whole system[2]. This new scenario is characterized by larger amounts of process data. Thus, data-driven solutions can provide major benefits in complex manufacturing processes [3-4].

As part of the H2020 SESAME project[2], the Friction Stir Welding (FSW) station of Pre-Final Assembly Line of the new Ariane 6 launcher is monitored. FSW [5] is raising as a key manufacturing technology in different high added value sectors. Emerging research is being carried out to characterize and understand the importance of the process parameters of the FSW process by means of Machine Learning (ML) approaches[5-6]. SESAME use case goes one step further due to the quality excellence required for the aerospace industry. Data-driven methods that ensure continuous quality estimation at each step of the process will bring the aerospace industry into the digital manufacturing paradigm, with a focus on the production Key Performance Indicators. In this scenario, process monitoring strategies that build upon ML can be applied to ensure quality estimation and zero-defect propagation along the assembly line. In this document, a preliminary analysis of the first experimental weldings is presented, as well as the

[1] Marta Camps, Eurecat, Centre Tecnològic de Catalunya, Applied Artificial Intelligence Dpt., Av. Carrer de Bilbao, 72, 08005 Barcelona, Spain; E-mail: marta.camps@eurecat.org.

[2] www.sesame-space.eu

convenience of the feature engineering layers to detect anomalies in the data that can be related to non-quality in the welded part.

2. Experimental Data

The Ariane 6 Pre-Final Assembly Line (P-FAL) is composed of two FSW stations, in charge of the longitudinal and circular weldings of the panels of the tanks. In this work, data from two different production batches of the circular station is presented. For each welding test, the evolution of 56 process variables is recorded. Different magnitudes are gathered for the different axis and subsystems of the welding station, presenting a diverging behaviour needed to be considered. The data analyzed is part of a preliminary study to find the optimal functioning point of the FSW stations. In the future, weldings will be performed with stable parameters and will enable further quality analysis.

3. Data segmentation: Regions of Interest

The principal strategy to tune ML models to estimate the quality and performance of the welding is to segment the data gathered into representative sections that can be then compared. Due to the high complexity of the assembly line, it is difficult to relate the dataset to a specific section or point in the welded part.. We propose to segment the welding into representative parts, following the structure of the circular FSW station.
Identifying this sections will enable comparisions between different experiments and sections, while building a feature engineering layer to define a common ground. The Tool Holder Temperature parameter (Figure 1) shows a pattern that allows identifying 16 sections, based on temperature spikes produced during the welding process.

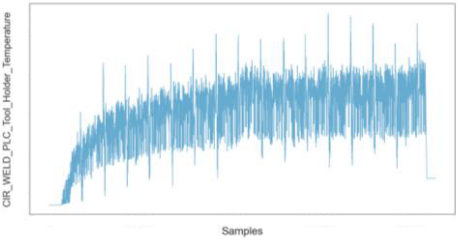


Figure 1. Welding dataset segmentation based on the Tool Holder Temperature behaviour.

4. Feature Engineering

To compare different experiments and different sections of the experiment, as they all differ in time duration and samples, equivalent variables must be set. To define a common ground and establish a fair comparison, the following set of features are proposed to capture the relevant information of each process parameter: min value, max value, mean value, standard deviation, number of samples, max value of the first order derivate, the two highest Power Spectral Density[7] and their corresponding frequencies. Comparing the proposed features for different experimental datasets and sections will allow identifying abnormal behaviors that can impact the welding quality. Figure 2 shows features values for the Spindle Temperature as a function of the section (x-axis) and the dataset file (colour). The different behaviour of section 11 can be observed.

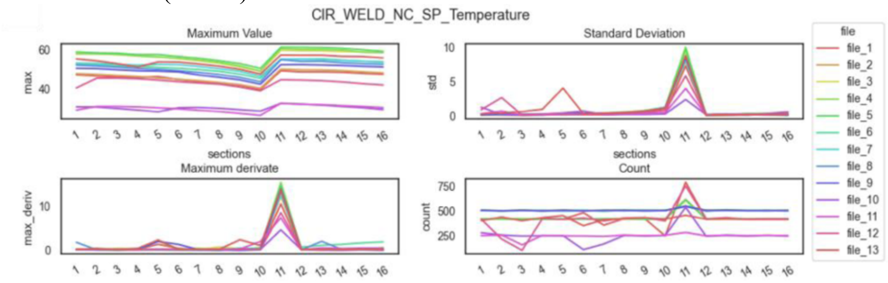


Figure 2. Example of features evolution for the Spindle Temperature parameter during the welding tests.

5. Complexity reduction and cross decomposition

Complexity reduction algorithms, such as Principal Component Analysis (PCA)[8], transform the problem by decreasing the data dimensionality, making the system more understandable for the process expert and encouraging their collaboration with ML systems.To compare different segments of the welding, a PCA projection is applied to the features. This analysis will help to identify clusters and abnormal behaviors. The presented results evaluate the different behavior of a given parameter among the 13 experiments and the 16 welding sections.

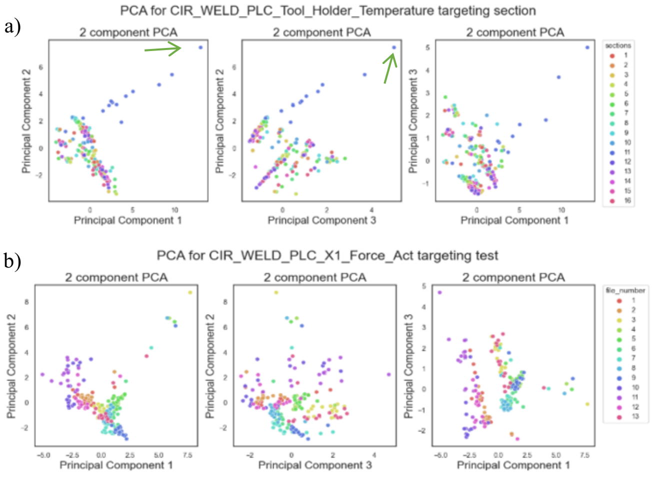


Figure 3. a) 3-PCA projection for the Tool Holder Temperature targeting welding sections. b) 3-PCA projection for the Force in X1 axis targeting experiments.

Figure 3a) presents the projection on the PCA space defined by sections for the Tool Holder Temperature, where the three components explain the 86,96% of data variance. It shows a clear cluster formed by section 11. This cluster is related to the lower temperature values that can be observed in Figure 1. However, one of the experiments behaves abnormally in section 11, as pointed by the green arrow. This abnormal behavior can affect the quality of the welding and provides valuable insight to the process expert to drive their quality inspection strategy of the welded tank. In Figure 3b) the three-component projection of the features of the Force in the X-axis can be seen. The selected components explain 83,45% of the information. Each color corresponds to a different welding and each dot a distinct section. Each weld creates separated clusters, indicating a strong dependency on the experiment. In addition, several outliers can be identified, most of them with higher values of the first and second principal components.

6. Discussion, conclusions and next steps

This paper presents the first results of the H2020 SESAME project regarding the data-driven predictive quality modules for the novel Friction Stir Welding station of the P-FAL of Ariane 6 launcher. A first analysis of the experimental data is described, focusing on the data preparation layer that will enable the future developments of the predictive quality modules. A feature engineering layer where several new features are created is introduced as well as a strategy for regions of interest segmentation that will allow the comparison of particular experiments and sections. The preliminary results already show the feasibility to identify abnormal behaviors in the welding process. Next steps will consider the correlation of the presented feature engineering strategy with the destructive inspection quality data, to feed supervised ML solutions: a classifier to determine the type of defect present in the welding and a regressor to estimate the severity of the defect.

7. Acknowledgment

This work has been carried out in the framework of the SESAME project, which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 821875. The content of this paper reflects only the author's view; the EU Commission/Agency is not responsible for any use that may be made of the information it contains.

References

[1] Zheng T, Ardolino M, Bacchetti A, Perona M. The applications of Industry 4.0 technologies in manufacturing context: a systematic literature review. Int. J. Prod. Res., 59:6, 1922-1954

[2] Reis MS, Gins G. Industrial Process Monitoring in the Big Data/Industry 4.0 Era: from Detection, to Diagnosis, to Prognosis. Processes. 2017; 5(3):35.

[3] Nieves J, Santos L, Bringas P G. Combination of machine-learning algorithms for fault prediction in high-precision foundries. Lect. Notes Comput. Sci., vol. 7447 LNCS, no. PART 2, pp. 56–70, 2012

[4] Susto G A, Schirru A, Pampuri S, McLoone S, Beghi A, "Machine Learning for Predictive Maintenance: A Multiple Classifier Approach," in IEEE Trans. Industr. Inform, vol. 11, no. 3, pp. 812-820, June 2015.

[5] Thomas W.M, Nicholas E.D, Friction stir welding for the transportation industries, Materials & Design, vol 18, Is 4–6, pp 269-273, 1997

[6] Nadeau F, Thériault B, Gagné M-O. Machine learning models applied to friction stir welding defect index using multiple joint configurations and alloys. P I MECH ENG L-J MAT , 234(5), 752–765

[7] Du Y, Mukherjee T, DebRoy T. Conditions for void formation in friction stir welding from machine learning. npj Comput Mater 5, 68 (2019)

[8] P. Welch. The use of the fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms, IEEE Trans. Audio Electroacoust. 2020. vol. 15, pp. 70-73, 1967.

[9] Pearson K. On Lines and Planes of Closest Fit to Systems of Points in Space. Philos. Mag. Lett. 1901. 2 (11): 559–572

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210133

We Are Not the Same Either Playing: A Proposal for Adaptive Gamification

Inmaculada RODRÍGUEZ [a], Anna PUIG [a] and Alex RODRÍGUEZ [a] [1]

[a] *Department de Matemàtiques i Informàtica, IMUB and UBICS Research Institutes, Universitat de Barcelona, Spain*

Abstract. Gamification consists in applying game mechanics in non-game contexts aiming at motivating and shaping behaviours. This paper proposes an adaptive approach for gamification, which takes as initial information players profiles - gathered from Hexad player type questionnaire - and considers also how these profiles change over time based on users interactions. Then, we provide the users with a personalised experience through the use of game elements that correspond to their dynamic playing profile. We present a preliminary evaluation of the approach by means of a simulator that yields promising results when comparing it with baseline configurations, i.e randomized and fixed player profile.

Keywords. adaptive gamification, dynamic player type

1. Introduction

Gamification applies elements of game playing to non-game contexts to pursue an objective while increasing user engagement and motivation. This objective varies widely depending on the context. For example, a gamified fair [21] was designed to provide visitants with the best experience visiting the fair. Gamification in MOOCs (Massive Open Online Course) aims to foster students' engagement and therefore increase the completion rate of courses [9]. In the bussiness context, the gamification of e-marketing processes aims to increase the site visits and sales [24].

Whatever the goal, gamification comes into the picture to encourage users behaviours through gameful experiences. Nevertheless, gamification designs usually take the one-fits-all approach, which may fail as result of considering that all users have the same profile. The opposite approach is adaptive gamification which takes into account userss diversity, i.e. the users are driven by different motivations.

Adaptive gamification found its basis on player types classifications. Nevertheless, whenever gamification has been approached from the adaptive perspective, it has been done gathering information about the user profile (i.e player types) fixed at the beginning of the process [16]. To do so, users answer a player type questionnaire, then the gamified system proposes the users game elements (e.g. badges, challenges) tailored to their pro-

[1] Research supported by Accio COMRDI18-1-0010, MISMIS-Language (PGC2018-096212-B-C33) and CI-SUSTAIN (PID2019-104156GB-I00).

files. However, these approaches have some limitations. This initial player profile either may have been provided by the user inaccurately or can slightly change/evolve along the time [5]. Additionally, adaptive gamified systems usually take the most predominant player type to show the user game elements related to this predominant player type, with the drawback of ignoring other player types that also may characterise the user, so that it should be considered several player types, not only the predominant one [13].

In this paper, we propose an adaptive approach for gamification to find the most suitable game elements for the player type of the user. Our approach uses initial players profiles, gathered from Hexad questionnaire [12], and further information about users interactions while using the system. Specifically, we consider user's interactions with game elements and user's opinions about those elements. Gamification goal is fostering users' engagement and so motivate the completion of online activities such as learning activities in a course or employees progress report in a company. Last but not least is to comment about the evaluation of the system. Although evaluations of the effects of gamification are commonly carried out with users, this paper presents a previous analysis of the proposed strategy using bots. The bots simulate different types of users, not so much to evaluate the effects of gamification (i.e. completion rate), but to validate the convergence and validity of our method.

2. Previous work

Recently, techniques user-centered have been proposed to correlate the game elements to different user profiles [12][14]. Some of them focused on specific user characteristics, such as motivation [4], personality treats [7] [8], learning styles [3], player types [16], or type of interaction with different activities [15]. Others combined different characteristics, such as [6] that took into account Learning Styles and Player Typesto end up determining the types of educational activities and game elements to include in a learning pathway. In contrast, [18] suggested using context, interactions, gender and player type to decide, through rules, the next game element to display. Other works focused on emotions to predict the individual's performance on gamified tasks, information that could potentially be used to adapt the game features [17].

Specifically, in current adaptive gamification approaches, the most commonly used taxonomies of player types have been Bartle's [2], BrainHex [20] and Hexad [1]. These taxonomies allow to identify player types easily from questionnaires and also allow to establish the correspondence between these Player Types and the Game Elements [14]. Questionnaires allow characterising the Player Type of a user with a set of ratings. They determine the player type prior to the experience. For example, the HEXAD model distinguishes between 6 player's types (Achiever, Player, Philanthropist, Disruptor, Socializer and Free Spirit), then as result of the questionnaire the user could obtain the ratings: Achiever 25%, Player 18%, Philanthropist 21%, Disruptor 8%, Socializer 10% and Free Spirit 5%. The decision of the final Player Type and thus, the most appropriate game element, usually relies on the predominant rating [16] or some combination of them [19]. In this work, we rely on the HEXAD player model and use the questionnaire proposed and validated by [23]. However, it should be noted that in all the studies analysed, the No-Player type of player is not considered, a fact that, in some cases, the use of gamification is more detrimental than beneficial [11]. Thus, we also add the No-Player player type to the HEXAD taxonomy, and also we consider for each user the assessment of all their ratings included in her Player Type.

Certainly, as supported by different studies [22], the interpretation of the results of these questionnaires may not be very reliable. Even if the questionnaire is valid, the answers may be somewhat random or the results at the beginning may not persistent during the experience depending on the moment or the mood the user. In fact, in addition to questionnaires, some proposals in the literature also gathered user feedback of learning activities [15], or scores on different game elements [10] during the experience. Thus, inspired by these works, we enrich the user model obtained from the initial questionnaire by means of user interactions and opinions during the course of the activity. Therefore, we will base the adaptation of the game mechanics to the "real" and "dynamic" user profile using two types of interactions: on the one hand, the interactions with the game elements and, on the other hand, the opinions that the user can make at certain moments about those elements. Thus, using both types of interactions, we will refine the players types during the experience, and, consequently, the game elements to be activated.

Regarding adaptive strategies proposed in the literature, most of them are more concerned with personalisation of learning rather than gamification per se [12]. In the following we refer to those that concretely focused on adapting gamification. [16] proposed a matrix factorisation model similar to those used in recommender systems. They used two matrices: one defining the player types of all users and the other representing how the game elements match the player types. They combined these two matrices to obtain game elements' scores for each user, and then they selected the element with the highest score. On the other hand, [6] defined an off-line Q-Learning algorithm to generate an adaptive learning path of the user. They defined S-Table and Q-Table that represented the correspondent state at each taken action, and the Q-values of each action in each state, respectively. Both tables were the same for all the profiles. Nevertheless, R-Table was specific for each Learning-Player profile. All of them consider adapting game elements to the initial player's profile, keeping the player' type static throughout the experience.

To the best of our knowledge our approach is the first that takes a dynamic picture of the user experience. Our adaptive algorithm is based on a matrix factorisation model similar to [16] that allows the recalculation of the player type, i.e. of all player ratings, during the experience in order to adapt the game element to the user model at any given moment. In this initial study, adaptation is based on activating one of the most appropriate game element at any time depending on the recalculated Player Type.

3. Runtime method for adaptive gamification

In this section we present our proposal for adaptive gamification. First, we introduce the main definitions our method is based on: (i) the Hexad player type model, including the non-player type, (ii) the selected game elements as a subset of those proposed by Tondello [23], and (iii) we define the matrices and vectors of ratings and interactions implied in our algorithm. Second, we describe how we match these player types with their corresponding game elements using an extended matrix factorisation method [16].

3.1. Previous definitions

Player type

An essential concept for adaptive gamification is the player type model, which classifies what kind of game elements maximise user motivation. As we mentioned above, we base on the Hexad player typology, adding the non-player type [1]. Player types are defined as:

$$PT = \{pt_1, pt_2, \ldots, pt_7\} \tag{1}$$

The player types, pt_i are explained below:

1. Disruptor: Motivated by the ability to modify the system.
2. Free Spirit: Motivated by the ability to freely explore the system.
3. Achiever: Motivated by the ability to win challenges and unlock hidden content.
4. Player: Motivated by the game itself.
5. Philanthropist: Motivated by the ability to share goods and help other users.
6. Socializer: Motivated by social connections.
7. Non-Player: Users who don't like to play.

The player type of a particular user is represented as a vector of length 7, PR, where each component r_i represents its ratings for each player type, so the values vary between 0 and 1, $PR = (r_1, r_2, \cdots, r_7)$. For example, a user can be 20% Disruptor, 10% Free Spirit, 30% Achiever, 40% Player, 40% Socializer, 10% Philanthropist and 0% No Player, which is encoded with the vector (0.2, 0.1, 0.3, 0.4, 0.4, 0.1, 0). Since the user's player type changes along the time, we define $PR^{(t)}$ as the Player Ratings at the time t such as:

$$PR^{(t)} = (r_1^{(t)}, r_2{}^{(t)}, \cdots, r_7{}^{(t)}) \tag{2}$$

Game element

Based on the correlation analysis of the Hexad player types with 52 game design elements done by [23], we select a subset of 14 game elements covering the whole spectrum of player types (see Figure 1). There are no game element associated to non-player type. Game elements are defined as:

$$GE = (ge_1, ge_2, \ldots, ge_{14}) \tag{3}$$

The game elements, ge_i, are briefly explained below:

1. Development Tool: Allows the player user to create some gamification mechanics such as badges, challenges and unlockables.
2. Challenge: The player must overcome a challenge, such as reaching a certain level, solving a problem in a certain time, etc.
3. Easter Egg: The mechanic consists of an image that, when pressed 5 consecutive times, allows access to a mini-game.
4. Unlockable: When a player overcomes a certain challenge, a hidden content is unlocked, which can be a message, a mini-game, etc.
5. Badge: Awarded to the player when she manages to complete a difficult task.
6. Level: Shows the user's progress in completing a task, subdivided into levels.
7. Point: The player gains score, experience, virtual money, etc.
8. Leaderboard: Displays a ranking of scores.

9. Gift Opener: The player opens gifts she has received.
10. Lottery: Game of chance (roulette) that allows to increase scores of a player.
11. Social Network: Small social network that allows players to create a profile, add friends and view their profile.
12. Social Status: Collection of rankings of players based on their scores, especially those related to social interaction, such as the number of followers, visitors, etc...
13. Share Knowledge: The player sends help messages to everyone in a group.
14. Gift: The player sends gifts to other users.

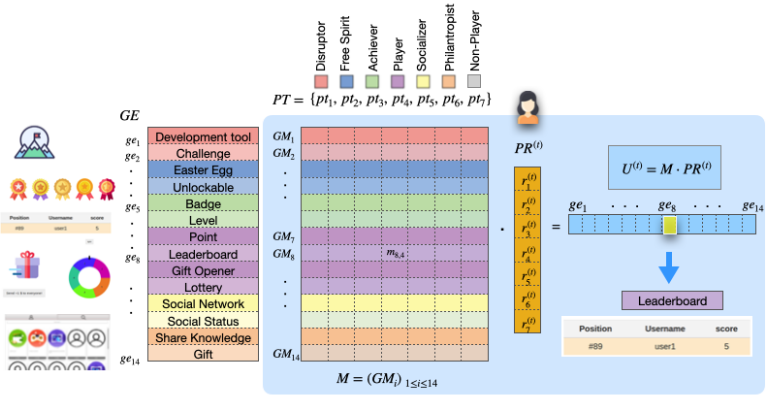


Figure 1. Selected 14 game elements, ge_i, enumerated by $i = 1..14$. Each colour represents one Player Type, pt_j, $(j = 1..7)$. The vector $PR^{(t)}$ defines the Player Ratings of a user, and the matrix M stores all the motivation values that the i-th game element produces to the j-th Player Type. The game element having the highest utility is the item that will be presented to the user. As an example, game element 8-th, the leaderboard, is highlighted to represent the next game element to be shown to the user.

It is worth mentioning that each game element does not target only one type of player, but can motivate different types of players. Thus, the i-th game element, ge_i, has associated a vector of motivation indexes GM_i, where each component, m_j, is the percentage of motivation it can cause in one of the 7 types of players, p_j. For example, the fifth game element ”Badge” (see Figure 1) can motivate users with both Achiever (pt_3) and Player (pt_4) player types, then the $GM_5 = (0,0,0.5,0.5,0,0,0,0)$.

$$GM_i = (m_1, m_2, \ldots, m_7) \;\; \forall i = 1\ldots 14 \tag{4}$$

Therefore, we define the matrix, M as:

$$M = (GM_i)_{1\leq i\leq 14} = (m_{ij})_{1\leq i\leq 14,\; 1\leq j\leq 7} \tag{5}$$

where the rows of this matrix are indexed by the game elements, and the columns by the player types. Thus, $m_{i,j}$ represents the percentage of motivation (motivation index) that the i-th game element produces to the j-th type of player.

Finally, as can be appreciated on the right side of Figure 1, the utility of using a game element, $U^{(t)}$, can be computed from the matrix M and the player type ratings, $PR^{(t)}$.

Interaction index

It is defined as the percentage of user's interaction with each game element at time t. This is represented by a vector of length 14 (the number of game elements), $S^{(t)} = (s_1^{(t)}, s_2^{(t)}, \cdots, s_{14}^{(t)})$. The interaction index of the i-th game element, $s_i^{(t)}$, is defined by

$$s_i^{(t)} = 1 - e^{-\left(o_i^{(t)} \frac{n_i^{(t)} - n_i^{(t-1)}}{\tau_i^{(t)} - \tau_i^{(t-1)}}\right)} \tag{6}$$

where,

$\tau_i^{(t)}$: the Display time i.e. the time interval for which the game element has been displayed until time t,
$n_i^{(t)}$: the number of interactions at time t,
$o_i^{(t)}$: the Opinion, i.e. user assessment about game element, it's a value between 0 and 1. Opinions from 1 to 5 stars correspond to 0.2, 0.4, 0.6, 0.8 and 1 respectively.

Note that Equation 6 encodes the interaction index as a number between 0 and 1. If there are no interactions between time t and $t+1$, the interaction index is 0 (since $(n_i^{(t)} - n_i^{(t-1)})$ is 0). The interaction speed is $(\frac{n_i^{(t)} - n_i^{(t-1)}}{\tau_i^{(t)} - \tau_i^{(t-1)}})$. Then, the interaction index tends to 1 as the interaction speed increases.

The Opinion $o_i^{(t)}$ modulates the interaction speed: $s_i^{(t)}$ tends faster to 1 when $o_i^{(t)}$ is near 1 than when $o_i^{(t)}$ is near 0.

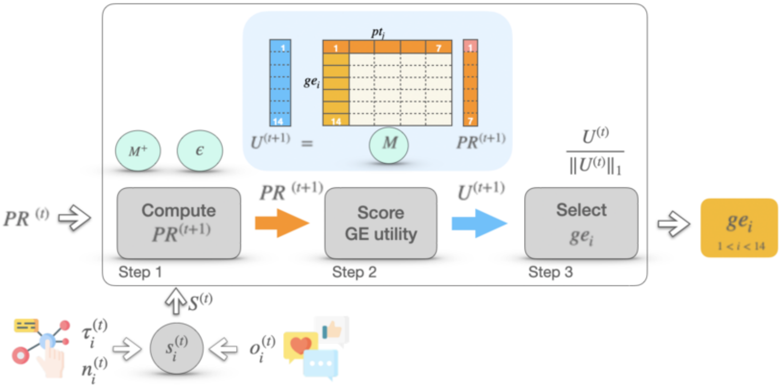


Figure 2. Steps to compute the utility of showing a ge_i associated to a user at time $t+1$. Blue circles indicate the constant data of our method. All other elements define dynamic values that change over time.

3.2. Adaptive method

We propose an iterative method that calculates the utility of showing a game element to a given user at time t. In each iteration, it performs three steps depicted in Figure 2 (considering definitions introduced in section 3.1). The steps are: (1) Obtain $PR^{(t+1)}$, (2) Score the utility of showing a game element to a user at a specific time $t+1$ (denoted by $U^{(t+1)}$), and (3) Select which game element to activate based on the assigned scores.

In the first step, we compute the new player type ratings, $PR^{(t+1)}$:

$$PR^{(t+1)} = (1-\varepsilon)\ PR^{(t)} + \varepsilon\ (M^{+} \cdot S^{(t)}) \tag{7}$$

where,

$PR^{(t)}$: the player type of the user at time t,
$S^{(t)}$: the interaction indexes,
ε: to avoid extreme fluctuations between $PR^{(t)}$ and $PR^{(t+1)}$, where $0 < \varepsilon < 1$. The value of this parameter should be tuned experimentally.
M^{+}: the Moore-Penrose pseudoinverse matrix of M, needed in order to interpret $S^{(t)}$ and $PR^{(t)}$ in the same space.

In the second, once we have calculated the new player profile, we compute the utilities as indicated in the top right side of Figure 1, using the matrix M defined in Equation 5:

$$U^{(t+1)} = M \cdot PR^{(t+1)} \tag{8}$$

Finally, in the third step, we select the next game element to display considering the i^{th} component of $\frac{U^{(t)}}{\|U^{(t)}\|_1}$ as the probability of choosing the i^{th} game element using a weighted random choice[2].

4. Simulations

An adaptive gamification strategy works if the game element proposed to the user fits its ”real” player profile at any time t. Considering that we simulate the player using a bot, in the following, we note the ”real” profile of the bot as RPT_0, and its player type rating at time t as $PR^{(t)}$.

The values of RPT_0 come from data gathered from user types Hexad test results[3], where 42782 tests were carried out, obtaining average type scores for all the modalities of players. We selected the eleven most representative modalities (see Table *Summary*[3])) and their corresponding average type scores (see Table *Average Type Scores*[3]). For instance, the Achiever modality appears in the 12% of the tests and its average scores are $RPT_0 = (0.12, 0.18, 0.20, 0.16, 0.16, 0.17, 0.0)$ meaning 12% Disruptor, 18% Free Spirit, 20% Achiever, 16% Player, 16% Socialiser, 17% Philanthropist, and 0% Non-Player.

Keeping in mind that a real user would answer the Hexad questionnaire reliably (accurately) or unreliably (inaccurately) at the beginning of the gamified experience, our bot simulates users' responses accurately or somewhat randomly. Therefore, we define $PR^{(0)}$ being close to RPT_0 ($RPT_0 \simeq PR^{(0)}$) or far away from ($RPT_0 \not\simeq PR^{(0)}$). To do so, if the reliability is low, the bot rating $PR^{(0)}$ is the furthest non-null scores from RPT_0 in Table *Average Type Scores*. Otherwise, if the reliability is high, the bot takes its $PR^{(0)}$ as RPT_0. Note that when the bot simulates accurate responses to the questionnaire it is desirable that the value of $PR^{(t)}$ remains close to RPT_0, while when it simulates inaccurate answers it is convenient that $PR^{(t)}$ converges to RPT_0 when $t \to \infty$.

[2] $\|\cdot\|_1$ is the ℓ_1-norm.
[3] https://gamified.uk/UserTypeTest2016/user-type-test-results.php

Besides, the bot interacts with the game elements obtaining interaction indexes, $S^{(t)}$, from $\tau_i^{(t)}$, $n_i^{(t)}$, $o_i^{(t)}$ (see input data of $s_i^{(t)}$ in the bottom left of Figure 2, and Equation 6). To do so, we use two variables: the time between two consecutive interactions, $\top_i^{(t)}$, and the opinion, $\Theta_i^{(t)}$, defined as:

$$\top_i^{(t)} = -9.5(GM_i \cdot RPT_0) + 10 \tag{9}$$

$$\Theta_i^{(t)} = \frac{1}{5}\,\texttt{round}\big(4(GM_i \cdot RPT_0) + 1\big) \tag{10}$$

where i corresponds to the game element (ge_i) selected by the method. Thus,

- If the game element (ge_i) fits the real bots profile (RPT_0), the bot interacts more frequently than otherwise. Therefore, $\top_i^{(t)}$ reflects this behaviour taking values from 0.5 (frequent interactions) to 10 (longer time between interactions).
- Regarding $n_i^{(t)}$, we consider that the bot interacts once every $\top_i^{(t)}$.
- The opinion can be calculated in a similar way using $\Theta_i^{(t)}$.

Once stated how the bot simulates a real user, we consider three experiment conditions, depending on whether of the value of the $PR^{(t)}$ is fixed or variable along the time, assuming the bot has answered the Hexad questionnaire previously (fixing $PR^{(0)}$). Moreover, we define how the error is computed in each case.

1. **Case A** - Constant $PR^{(0)}$: A constant player rating, $PR^{(t+1)} = PR^{(0)}$, is assigned at any time. In this case, we use Equation 8 to calculate $U^{(t+1)}$ once and always use its maximal component.
   We calculate the error as follows. Since $PR^{(t+1)} = PR^{(0)}$, the distance is simply calculated as $Err = |RPT_0 - PR^{(0)}|$. Note that if the user has answered the questionnaire accurately, we have $Err = 0$.
2. **Case B** - Random Dynamic $PR^{(t)}$: A dynamic player rating $PR^{(t+1)}$ is randomly chosen at each time t and then, the game element selected to be shown is also the maximal component of $U_i^{(t+1)}$. Note that this case iW equivalent to pick a random game element.
   In this case, we compute the error as the average distance between RPT_0 and a random point $p \in \{(x_1,\ldots,x_7) \in [0,1]^7 : \sum_{i=1}^{7} x_i = 1\}$ using the Average Distance of Random Points in a Unit Hypercube[4].
3. **Case C** - our method, Dynamic $PR^{(t)}$: A dynamic player rating $PR^{(t+1)}$ is computed according the $s_i^{(t)}$ defined by Equation 6. Then, the bot simulates $\tau_i^{(t)}$, $n_i^{(t)}$, and $o_i^{(t)}$ using $\top_i^{(t)}$ and $\Theta_i^{(t)}$. The game element selected to be shown is a weighted random choice of $U^{(t+1)}$ (see step 3 in Figure 2).
   We calculate the error based on the distances between $PR^{(t)}$ and RPT_0 for all t from 1 to n_iter, $Err = \frac{1}{n\_iter}\sum_{t=1}^{n\_iter} |RPT_0 - PR^{(t)}|$.

Experiments were performed using a homemade gamification sofware and run on Windows 10, Intel Core i7 processor with 8GB RAM. Table 1 shows the mean errors

[4] Average Distance of Random Points in a Unit Hypercube. https://martin-thoma.com/curse-of-dimensionality/

obtained in all the cases. In case A, as we take into account $PR^{(0)} = RPT_0$, accurate answers is the ideal case ($Err = 0$). However, with inaccurate ones the error grows to 0.0311. Moreover, both cases B have similar errors (0.08024 and 0.08027), because is a random selection of player type ratings and thus, they are almost independent of the reliability (accurate or inaccurate) of the answers. Finally, except in case A with accurate answers, case C behaves better than A and B even in its worse scenario ($Err = 0.0333$).

Case	**A** Constant $PR^{(0)}$	**B - Random** Dynamic $PR^{(t)}$	**C - Our method** Dynamic $PR^{(t)}$
Accurate answers: $RPT_0 \simeq PR^{(0)}$			
Mean (SD)	0	0.08024 (0.00052)	0.0070 (0.00166)
Worse scenario of C	0	0.0804	0.0105
Best scenario of C	0	0.0797	0.0029
Inaccurate answers: $RPT_0 \not\simeq PR^{(0)}$			
Mean (SD)	0.0311(0.00404)	0.08027 (0.00040)	0.0243 (0.00475)
Worse scenario of C	0.0367	0.08012	0.0333
Best scenario of C	0.0233	0.08018	0.0146

Table 1. Table of mean errors, Err, when the questionnaire is answered accurately or inaccurately), together with the best and worst errors obtained in case C.

In summary, we conclude that the case A - where player type ratings are fixed along the experience - is the best when the users answer the questionnaire thoroughly and accurately, while our method, case C, works well in both cases (respond it either accurately or inaccurately). In fact, since it won't be possible to know whether (real) users answer the player type questionnaire accurately or not in real situations, our method is the most suitable for an adaptive gamified experience.

5. Conclusions

This research proposes a method to present the users game elements that fit their profile (player type). However, instead of taking a (static) picture of the profile at the beginning of the experience, we consider how it may change along the course of gamified activities. The method calculates the utility of showing a game element to the user based on the evolved Player Type (PT), the interactions on game elements, and the scores given by the users to those game elements. A bot simulated three different cases: constant player type ratings, random dynamic player type ratings, and dynamic player type ratings, where player type is recomputed using player's interactions, opinions at each step of the iterative method. The results show that our method achieves a low error considering both situations: when the user answers the player type questionnaire accurately ($Err = 0.0070$) and inaccurately ($Err = 0.0243$). As future work we will test the method with users and incorporate new inputs to our method such as emotions and activities completion.

References

[1] M Andrzej. Even Ninja Monkeys Like to Play: Gamification, Game Thinking and Motivational Design, 2015.

[2] Richard A Bartle. Hearts, Clubs, Diamonds, Spades: Players Who Suit Muds Want more papers like this? *Journal of MUD research*, 1(1):19–undefined, 1996.

[3] S. Borges, R. Mizoguchi, V. H S Durelli, I. Bittencourt, and S. Isotani. A link between worlds: Towards a conceptual framework for bridging player and learner roles in gamified collaborative learning contexts. In *Advances in Social Computing and Digital Education*, pages 19–34. Springer, 2016.

[4] G. C. Challco, D. A. Moreira, R. Mizoguchi, and S. Isotani. An ontology engineering approach to gamify collaborative learning scenarios. In *CYTED-RITOS International Workshop on Groupware*, pages 185–198. Springer, 2014.

[5] D Charles and M Black. Dynamic Player Modelling: A Framework for Player-centred Digital Games. *Proceedings of 5th International Conference on Computer Games: Artificial Intelligence, Design and Education (CGAIDE'04)*, Microsoft(April):29–35, 2004.

[6] E. Chtouka, W. Guezguez, and N. B. Amor. Reinforcement learning for new adaptive gamified LMS. In *International Conference on Digital Economy*, pages 305–314. Springer, 2019.

[7] M. Denden, A. Tlili, F. Essalmi, and M. Jemni. Does personality affect students' perceived preferences for game elements in gamified learning environments? In *Proceedings - IEEE 18th International Conference on Advanced Learning Technologies, ICALT 2018*, pages 111–115, 2018.

[8] L. S. Ferro, S. P. Walz, and S. Greuter. Towards personalised, gamified systems. pages 1–6, 2013.

[9] O. Gené, M. Núñez, and A. Blanco. Gamification in MOOC: Challenges, opportunities and proposals for advancing MOOC model. *ACM International Conference Proceeding Series*, pages 215–220, 2014.

[10] S. Guggiari. Emergent Personalized Content in Video Games. (April), 2019.

[11] S. Hallifax, E. Lavoué, and A. Serna. To tailor or not to tailor gamification? An analysis of the impact of tailored game elements on learners behaviours and motivation. *Lecture Notes in Computer Science*, 12163 LNAI:216–227, 2020.

[12] S. Hallifax, A. Serna, J. Marty, and E. Lavoué. Adaptive Gamification in Education: A Literature Review of Current Trends and Developments. *Lecture Notes in Computer Science*, 11722 LNCS:294–307, 2019.

[13] S. Hallifax, A. Serna, J. Marty, G. Lavoué, and E. Lavoué. Factors to consider for tailored gamification. *CHI PLAY 2019 - Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, pages 559–572, 2019.

[14] A.C.T Klock, I. Gasparini, M.S. Pimenta, and J. Hamari. *Tailored gamification: A review of literature*, volume 144. 2020.

[15] A. Knutas, J. Ikonen, D. Maggiorini, L. Ripamonti, and J. Porras. Creating student interaction profiles for adaptive collaboration gamification design. *International Journal of Human Capital and Information Technology Professionals (IJHCITP)*, 7(3):47–62, 2016.

[16] . Lavoué, B. Monterrat, M. Desmarais, and S. George. Adaptive Gamification for Learning Environments. *IEEE Transactions on Learning Technologies*, 12(1):16–28, 2019.

[17] C. Lopez and C. Tucker. Towards personalized adaptive gamification. 2018.

[18] B. Monterrat, E. Lavoué, and S. George. Toward an Adaptive Gamification System for Learning Environments. In *Computer Supported Education*, pages 115–129, Cham, 2015. Springer International Publishing.

[19] A. Mora, G. F. Tondello, L. E. Nacke, and J. Arnedo-Moreno. Effect of personalized gameful design on student engagement. *IEEE Global Engineering Education Conference, EDUCON*, 2018-April:1925–1933, 2018.

[20] L. E. Nacke, C. Bateman, and R. L. Mandryk. BrainHex: A neurobiological gamer typology survey. *Entertainment Computing*, 5(1):55–62, 2014.

[21] A. Rapp, M. Alessandro, R. Simeoni, L. Console, and others. Playing while Testing: How to Gamify a User Field Evaluation. In *Designing Gamification: Creating Gameful and Playful Experiences held in conjunction with SIGCHI Conference on Human Factors in Computing Systems (CHI'13)*. Gamification Research Network, 2013.

[22] J. Sabourin and J. Lester. Affect and Engagement in Game-BasedLearning Environments. *IEEE Transactions on Affective Computing*, 5(1):45–56, 2014.

[23] G. F. Tondello, R. Wehbe, L. Diamond, M. Busch, A. Marczewski, and L. E. Nacke. The Gamification User Types Hexad Scale Gustavo. *CHI PLAY 2016 - Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play*, pages 229–243, 2016.

[24] DM-H Wen, D-J-W Chang, Y-T Lin, C-W Liang, and S-Y Yang. Gamification design for increasing customer purchase intention in a mobile marketing campaign app. In *International conference on HCI in business*, pages 440–448. Springer, 2014.

Machine Learning

This page intentionally left blank

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210135

Combining Simulations and Machine Learning for Efficient Prediction of Process Parameters Evolution in Injection Moulding

Albert ABIO [a,1], Francesc BONADA [a] and Oriol PUJOL [b]
[a] *Eurecat, Centre Tecnològic de Catalunya, Unit of Applied Artificial Intelligence, Cerdanyola del Vallès, Spain*
[b] *Department de Matemàtiques i Informàtica, Universitat de Barcelona, Barcelona, Spain*

Abstract. In recent years, the emerging technologies in the context of Industry 4.0 have led to novel approaches in process monitoring and control, such as the introduction of Reinforcement Learning and Digital Twins. Consequently, large amounts of data, precise modelling and exhaustive simulations are required. The aim of this work is to propose a methodology based on the technique of backward selection to reduce the number of reference points in the simulation stage of manufacturing processes, enhancing the efficiency of data generation and the simplicity of the simulations. The methodology is proved in the particular case of plastic injection moulding simulations.

Keywords. Industry 4.0, Backward Selection, Machine Learning, Manufacturing Simulation, Plastic Injection Moulding, Node Reduction

1. Introduction

Simulations are widely used in several fields, but one of their most relevant applications is in manufacturing, where they are established as a powerful tool for evaluating, validating and optimizing design and manufacturing processes [1]. Additionally, they also allow the digitalization of the manufacturing system, being able to provide knowledge without perturbing the real system.

The current manufacturing paradigm is drifting towards data-driven systems. These define the core of what is known as Industry 4.0. Under this paradigm, new tools and techniques based on data and artificial intelligence are introduced as drivers for innovation that impact in the productivity. The treatment with advanced analytic techniques of the information acquired during the manufacturing process allows gathering complex and precise knowledge about the system as well as to enable the deployment of predictive quality and maintenance protocols [2]. Machine Learning (ML) techniques have in-

[1] E-mail: albert.abio@eurecat.org

creased their relevance to address manufacturing problems, since they can deal with high dimensional data and they are able to uncover and model nonlinear relationships [3].

Supervised and unsupervised ML methods have been the most usual techniques applied in manufacturing to date, although Reinforcement Learning (RL) has started to show its high potential in this field [4].

The Digital Twin (DT) is one of the latest technology trends in the framework of Industry 4.0. It consists in a digital version of a physical entity that exists in a virtual space and that it is in constant interaction with the physical space [5]. At present, there is a strong interest in the research and development of DTs, since they are able to use the information of a real-world system into a virtual system, enhancing the knowledge of the operating status and optimizing its performance.

The implementation of DTs requires difficult modelling through exhaustive simulations. Moreover, to effectively apply ML techniques, and in particular RL strategies, a large amount of data needs to be generated. In the particular case of manufacturing, the experimental data is costly and involves the waste of raw materials, making the acquisition of data under non-productive conditions difficult or unfeasible. In this scenario simulations become the main source of data. However, generating data using simulations models of the manufacturing processes is time consuming and computationally expensive. The direct consequence is the difficulty in the development of DTs or the application of RL in manufacturing. A possible solution to this problem is to combine simulations with ML predictive techniques to enhance and speed up the virtual data generation.

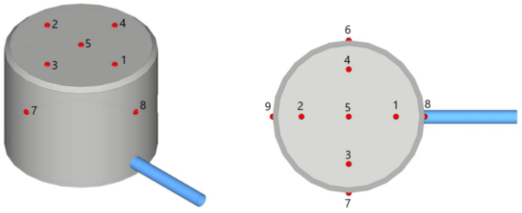
The approach of combining simulations and ML can be used to build simpler physical models, which reduce the computational resources that complete models require. These kind of models are usually called surrogate models or response systems, which represent the system in a simpler but representative way. Hence, the data is obtained efficiently without a relevant lost of knowledge about the system [6].

In this work, we explore the hybridization of physical phenomenological simulations with ML prediction techniques and we develop a methodology to increase the simulation efficiency, in the particular case of plastic injection moulding. The goal of this hybrid model is to combine ML predictions with a reduced number of simulation nodes with the goal of describing a more complete phenomenon. In the proposed study the best points to simulate in order to obtain a reliable description of the process are identified. Additionally, an adapted backward selection methodology is used for node selection task.

2. Plastic Injection Moulding Simulation

Plastic injection moulding consists in the injection of melted plastic into a mould where it cools down and solidifies to acquire the desired shape [7]. The process comprises different steps: First, the plastic is melted inside a barrel applying temperature, pressure and shear using a rotational screw. Then, the injection of the required plastic into the mould is carried out through a shot. Afterwards, a packing pressure is applied to guarantee part dimensional characteristics.

In this study, the objective is to increase the efficiency of data generation in plastic injection moulding process. For this reason, simulations of the process have been performed with the Moldex3D mold flow analysis commercial software [8], based on Finite Element Analysis (FEA). The studied part is a cap injected in a mould cavity. In Figure

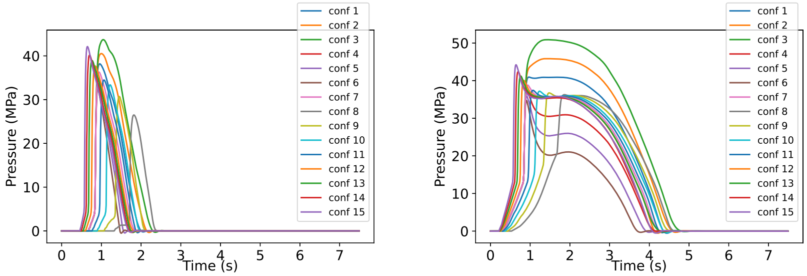


(a) (b)

Figure 1. Sketch of the cap form different perspectives. The melted plastic enters through the blue runner. The red dots are the sensed points and the numbers are used to identify the different pressure sensors.

Table 1. Values of the packing pressure (PP) and the injection speed (v) for the 15 generated configurations. The marked configurations (*) are used for testing.

Configuration	$PP(MPa)$	$v(mm/s)$	Configuration	$PP(MPa)$	$v(mm/s)$
conf 1	$PP_{ref}+0.1PP_{ref}$	v_{ref}	conf 9	PP_{ref}	$v_{ref}-6$
conf 2*	$PP_{ref}+0.2PP_{ref}$	v_{ref}	conf 10	PP_{ref}	$v_{ref}-4$
conf 3	$PP_{ref}+0.3PP_{ref}$	v_{ref}	conf 11*	PP_{ref}	$v_{ref}-2$
conf 4	$PP_{ref}-0.1PP_{ref}$	v_{ref}	conf 12	PP_{ref}	$v_{ref}+2$
conf 5*	$PP_{ref}-0.2PP_{ref}$	v_{ref}	conf 13	PP_{ref}	$v_{ref}+4$
conf 6	$PP_{ref}-0.3PP_{ref}$	v_{ref}	conf 14*	PP_{ref}	$v_{ref}+6$
conf 7	PP_{ref}	v_{ref}	conf 15	PP_{ref}	$v_{ref}+8$
conf 8	PP_{ref}	$v_{ref}-8$			



(a) (b)

Figure 2. Pressure evolution in the simulation of the process for all the configurations in (a) SN5 and (b) SN8.

1, the geometry of the cap is illustrated. Moreover, nine virtual or simulated sensors have been displayed in the geometry: SN1 - SN9. The sensors measure the cavity pressure evolution in these nine selected points, since it has been identified in the literature as one of the most relevant variables for the quality of the final product [7].

To extend the study to different conditions, several configurations have been generated changing two parameters of the simulations: the injection speed (v) of the plastic into the mould and the value of the packing pressure (PP), as shown in Table 1. These parameters have been modified around their nominal working values v_{ref} and PP_{ref}. The result are 15 configurations representing the conditions of a real-environment. The simulations last in the range of 40 to 50min per configuration.

The output of the simulations is the evolution of the cavity pressure exerted by the melted plastic in the nine points where the sensors are located. Initially, the sampling time in Moldex3D for the pressure data is different for each simulated configuration. In order to homogenize them, we apply a resampling each $0.01s$ to homogenize all the configurations, obtaining 750 samples per configuration. In Figure 2, the pressure evolution behavior is shown for all the different configurations in two of the sensed points. It is worth noting the heterogeneity of the results for different sensors and configurations.

3. Experimental Setup and Methodology

In this section, we present the experimental set-up that aims to understand up to which extend simulation nodes can be replaced with ML predictions. To this end, two experiments are carried out: An individual assessment of each sensor and a global assessment for the complete set of sensors. In this article, we adopt a methodology that systematically replaces simulated sensors by predicted ones by means of a backward search. In particular, the sensors selected for testing purposes are such that they define the worst case scenario, ensuring that any other choice would achieve better scores.

3.1. Experimental Setup

From the nine different simulated sensors, three of them will be used for assessing the quality of the prediction system. The three selected target pressure sensors will be predicted using the remaining simulated sensors data. Additionally, to ensure the generalization of the algorithm on independent test data, the set of 15 configurations is split in 11 configurations for training and a 4 configurations for testing (see values in Table 1 marked with a star). The test configurations have been chosen to be intermediate values of the simulation parameters (injection speed and packing pressure).

To select the target testing sensors, we use the concept of similarity between the samples of the time series of the pressure sensors. Then, we compute the mean similarity [11] between the pressure curves of the sensors $\mathrm{tsim}(X,Y)$ as follows,

$$\mathrm{tsim}(X,Y) = \frac{1}{n}\sum_{i=1}^{n} \mathrm{numSim}(x_i,y_i) \tag{1}$$

where $X = x_1,...,x_n$ and $Y = y_1,...,y_2$ are time series of two pressure sensors and $\mathrm{numSim}(x_i,y_i) = 1 - \frac{|x_i-y_i|}{|x_i|+|y_i|}$ is the similarity between two samples in the same instant of time. The operation range of $\mathrm{tsim}(X,Y)$ lies in the interval $[0,1]$. $\mathrm{tsim}(X,Y) = 1$ refers to two identical pressure curves.

Figure 3 is the result of computing the mean similarity between the different sensors averaging for the 4 test configurations. This result drives the selection of the less similar sensors for the prediction, with the intention to use non-trivial cases to validate the presented methodology.

Comparing Figure 3 with the localization of the sensors in the cap (Figure 1), we can observe the symmetry relations displayed in the Mean Similarity Matrix between the sensors SN3 and SN4 and the sensors SN6 and SN7. These sensors will be discarded for prediction because they will not suppose a difficulty for the algorithm, that will use the corresponding symmetric sensor to obtain a very good prediction. The remaining

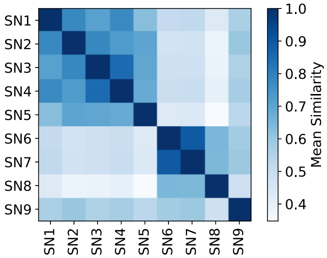


Figure 3. Mean Similarity Matrix averaged over the 4 test configurations. The spatial regions of the cap can be differentiated in this matrix. The first 5 sensors are in the superior part of the cap and the last 4 in the lateral.

sensors are not symmetric due to the position of the runner. The three sensors with less similarity are SN5, SN8 and SN9. Due to the central position of SN5 in the cap, we are interested in the real value of the cavity pressure in that point. We prefer not to include SN5 in the set of target sensors and replace it with SN2, which has no symmetries in the cap geometry. Summarizing, in a first approach, the sensors SN2, SN8 and SN9 will be predicted using the values of the rest of pressure sensors. As mentioned, this defines a worst case scenario.

3.2. Backward Selection Methodology

In the previous section we have defined a set of three target sensors that will be predicted using the data from the six remaining sensors. This means that in future simulations six points will still have to be sensed. In order to minimize the number of sensed points for future simulations and explore to which extend these can be replaced by ML predictions we propose to use a methodology based on the technique of backward selection [12].

The technique consists in the elimination of the input features of a ML algorithm, using a metric that allows to decide which feature is the best to drop in a greedy manner. Starting from a set of $k = 1,...,M$ input features and $i = 1,...,L$ test configurations, the elimination of features is carried out through the following iterative process:

1. Use the current number of features M to predict the target.
2. Compute the error metric for each of the test configurations, the Mean Squared Error (MSE) in our case.
3. For all the $k = 1,...,M$ current features:
   (a) Predict the target without using the feature k.
   (b) Compute the error metric for all the test configurations.
   (c) Calculate $\mathrm{diff}_i^{M-1,M}(k)$, the error difference with and without feature k for each test configuration as follows,

$$\mathrm{diff}_i^{M-1,M}(k) = \mathrm{MSE}_i^{M-1}(k) - \mathrm{MSE}_i^{M} \tag{2}$$

   (d) Perform the weighted average $t_{conf}^{M}(k)$ described in the following Eq. (3) of the differences over the test configurations for the eliminated feature k.

$$t_{conf}^{M}(k) = \sum_{i=1}^{L} c_i(k)\,\mathrm{diff}_i^{M-1,M}(k) \tag{3}$$

where $c_i(k) = \frac{\mathrm{MSE}_i^M}{\sum_{j=1}^{L} \mathrm{MSE}_j^M}$

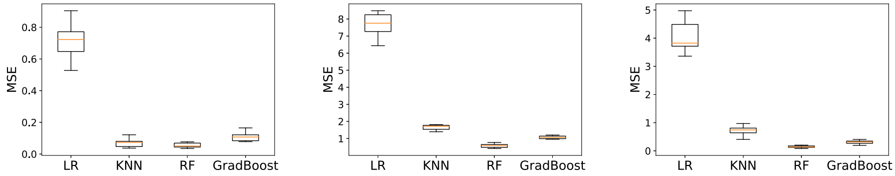
4. Select the smaller value of $t_{conf}^M(k)$ and drop the corresponding feature k.
5. Repeat the process with the new set of features of size $M = M - 1$.

The process ends when the number of desired features is reached or when the error overcomes a given threshold. Observe that the value of $\mathrm{diff}_i^{M-1,M}(k)$ may be negative if the elimination of the feature k improves the prediction algorithm performance. The proposed methodology takes into account the value of error metric for each test configuration to decide which is the best feature to drop, since it is preferable to optimize the predictions of the configurations that have a higher error.

4. Results and Discussion

4.1. Baseline Prediction Results

The reduction of the sensed points is realized with the application of a ML regression algorithm that uses as input data coming from a few locations to predict the rest of the points. Before that, we perform an algorithm comparison in order to know the prediction capability of some regression algorithms to all the available data. Therefore, we will randomly merge the data from all the configurations, obtaining a dataset composed by 750×15 samples and 9 features. Selecting a target sensor to predict and using all the others for training, we will implement a 10-Fold CV [9] to choose a candidate algorithm.



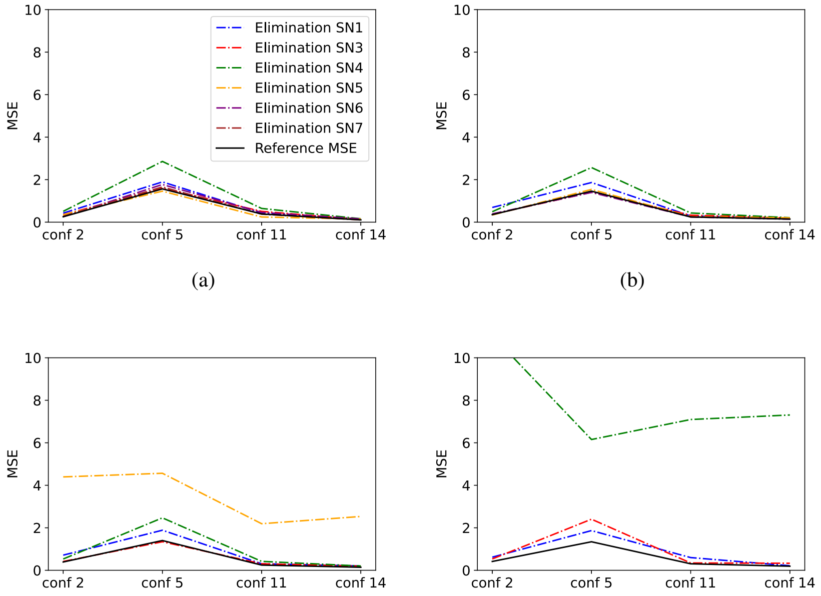
(a) (b) (c)

Figure 4. Average Mean Squared Error (MSE) of a 10-Fold CV for algorithm comparison between Linear Regression (LR), k-Nearest Neighbours Regressor (KNN), Random Forest Tree Regressor (RF) and Gradient Boosting Regressor (GradBoost). Target predicted sensor: (a) SN2. (b) SN8. (c) SN9.

Figure 4, shows the error performance comparison of four different regression techniques applied to the complete dataset. Random Forest Tree Regressor [10] achieves a lower error rate and will be used for the rest of the experiments.

4.2. Individual Sensor Reduction Assessment

The purpose of this study is to demonstrate the feasibility of achieving an important reduction of the number of sensed points without having a high impact in the prediction error. As explained in the experimental setup section, we will reduce the number input sensors used to predict the set of three target sensors, by applying the methodology presented in section 3.2.



(a) (b)

(c) (d)

Figure 5. MSE in test configurations for target SN9. The black curve indicates the reference MSE from the previous step of the elimination process and the colored curves indicate the MSE with the drop of one of the sensors. (a) 5 sensor selection. (b) 4 sensor selection. (c) 3 sensor selection. (d) 2 sensor selection.

The different steps of the backward selection process are displayed in Figure 5, where the MSE for each test configuration is represented when we eliminate the input sensors. The curve of reference MSE does not suffer a relevant variation during the stages of the process, meaning that the prediction capability of the algorithm remains despite the discarded sensors. It refers to the MSE computed with M sensed points and it is used to evaluate the predictions with $M-1$ sensed points through the use of Eq. (3).

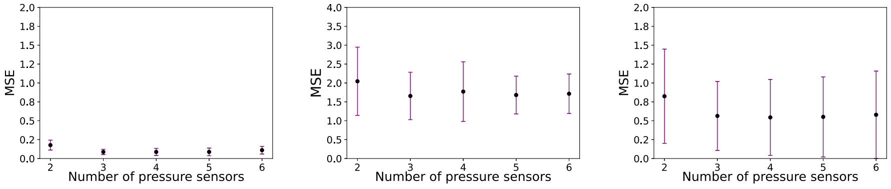
Figure 6 shows the result of the selection process for each one of the target sensors. The evolution of the mean MSE of the 4 test configurations allows to identify a threshold in three input sensors. Below this threshold, the use of less input sensors induces the error metric to start having a relevant increase. By inspecting these values, Table 2 shows the three best input sensors for individually predicting each target sensor.

Table 2. Best input sensors for the corresponding target sensors.

Target sensor	**SN2**	**SN8**	**SN9**
Best input sensors	SN1, SN3, SN6	SN6, SN1, SN5	SN6, SN4, SN1

4.3. Global Sensor Reduction Assessment

The results in the previous section 4.2 open the possibility of reducing the number of sensed points, showing that each individual sensor can be predicted using 3 sensed points



(a) (b) (c)

Figure 6. Evolution of the mean and the standard deviation over configurations with the number of input sensors used in the prediction. (a) SN2. (b) SN8. (c) SN9.

without a relevant effect in the error. However, the results show that different nodes require of different sensors for a good performance. In this subsection we consider whether a small common set of sensors may suffice for predicting all targets.

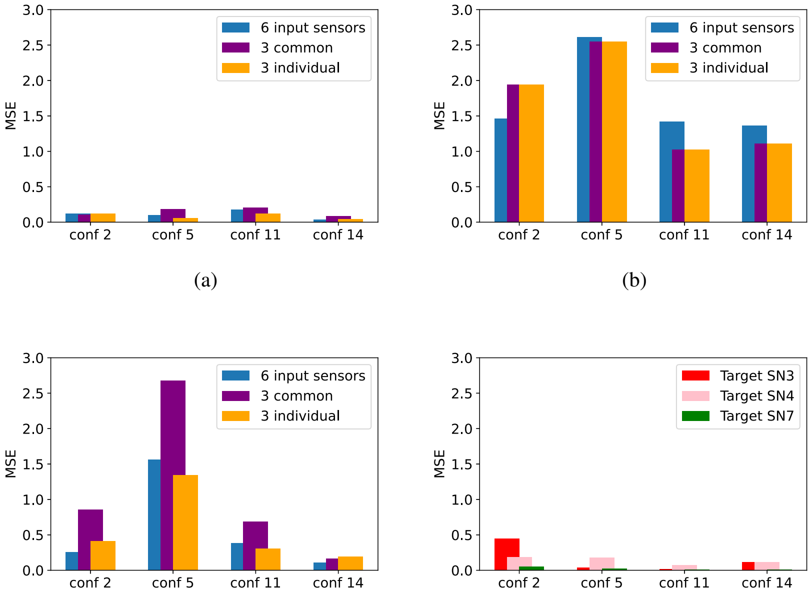
In order to do so, we will select the most repeated input sensors to predict all the target sensors. With the information of Table 2, we can identify that SN1 and SN6 are important for the prediction of the 3 target sensors. Additionally, we will choose the SN5, since it has a central position in the cap geometry (Figure 1). Accordingly, the final set of input sensors is formed by SN1, SN5 and SN6.

The defined final set of input sensors is used to predict the pressure of the target sensors. In Figure 7, the error metric MSE is compared when the prediction is done with 6 or 3 input sensors. If we use the individual set of 3 sensors of Table 2, we achieve a decrease of the error in most of the cases. Elseways, the use of the common set of sensors leads to a higher prediction error, but it allows to reduce the number of sensed points in the simulations. Moreover, the common set of sensors is not only able to predict the target sensors but also it yield good predictions for all the remaining sensors that the methodology has discarded. Figure 7d shows these results and it demonstrates the generalization capability of the proposed methodology. In the framework of industrial problems, it can be useful to include previous knowledge in a human - AI interaction that aims to help the global system performance, as shown including SN5 in the common set of input sensors.

As a final result, Figure 8 shows simulated and predicted pressure curves of the target sensors for a certain configuration. As observed in Figure 7, the variability of the accuracy of the prediction is highly dependent on the target sensor and the test configuration. Figure 8c shows the worst case prediction. Regardless of not being a perfect prediction, some relevant features of the injection moulding process such as the maximum value of the curve or the duration of the different stages are correctly characterized [7]. This result is of high importance as in manufacturing processes, the control a few relevant process variables is enough to determine the global performance of the system.

5. Conclusions

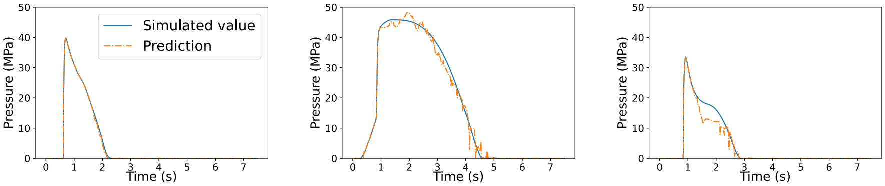
In this work, we have studied the hybridization of simulations with ML predictions applied to increase efficiency in a plastic injection moulding simulation process. The main results show that sensor nodes can be replaced with predicted versions using a very small set of real simulated data. This has been tested in experiments where process parameters



(a) (b)

(c) (d)

Figure 7. MSE comparison using 6 input sensors, the final set of 3 input sensors or the best 3 selected sensors for each individual target sensor. Target sensor: (a) SN2. (b) SN8. (c) SN9. (d) MSE of the prediction of the sensors SN3, SN4 and SN7 as target with the selected set of 3 input sensors.



(a) (b) (c)

Figure 8. Comparison example between the simulated and the predicted temporal evolution of the pressure using the 3 selected input sensors. (a) Target SN2, conf 14. (b) Target SN8, conf 2. (c) Target SN9, conf 5.

(packing pressure and injection velocity) are different from those used for the data in the training set. Although in specific configurations the results may be worsened due to the reduction in the sensed nodes, the predictions obtained still preserve the critical process variable values, namely, maximum pressure value, duration of the process stages, etc. This is an important result because the global performance of the system is highly dependent on these values.

Although the proposed methodology has been used for a particular manufacturing problem, it can be extended to other applications. For instance, in the generation of the experimental data, it may be interesting to use less sensors to obtain the same amount of data due to the economical impact of these devices. On the other hand, in the simulation field, the adaptation of the methodology for a more general scenario with a larger number of points to reduce may still be a challenge to address. The generation of more complex similar surrogate models could have a high impact in the efficiency of simulations. The decrease of the nodes used for simulation could lead to lower simulation times, which are an issue in some common simulation methods like Finite Element Method (FEM).

Finally, future research focus on increasing the efficiency of the data generation by means of more complex surrogate models that are based on the hybridization of simulations and ML. In the context of Industry 4.0, this will immediately boost the development of promising data-driven technologies that need a large quantity of data to be implemented, such as the DTs or RL.

Acknowledgements

Albert Abio is a fellow of Eurecat's ”Vicente López” PhD grant program. This work has been partially funded by the Spanish project PID2019-105093GB-I00 (MINECO/ FEDER, UE).

References

[1] Mourtzis D, Doukas M, Bernidaki D. Simulation in Manufacturing: Review and Challenges. Procedia CIRP. 2014; 25: 213-229.

[2] Tao F, Qi Q, Liu A, Kusiak A. Data-driven smart manufacturing. Journal of Manufacturing Systems. 2018; 48: 157-169.

[3] Wuest T, Weimer D, Irgens C, Thoben KD. Machine learning in manufacturing: advantages, challenges, and applications. Production and Manufacturing Research. 2016; 4(1): 23-45.

[4] Oliff H, Liu Y, Kumar M, Williams M, Ryan M. Reinforcement learning for facilitating human-robot-interaction inmanufacturing. Journal of Manufacturing Systems. 2020; 56: 326-340.

[5] Tao F, Zhang H, Liu A, Nee, AY. Digital twin in industry: State-of-the-art. IEEE Transactions on Industrial Informatics. 2018; 15(4): 2405-2415.

[6] von Rueden L, Mayer S, Sifa R, Bauckhage C, Garcke J. Combining Machine Learning and Simulation to a Hybrid Modelling Approach: Current and Future Directions. International Symposium on Intelligent Data Analysis. 2020: 548-560.

[7] Kurt M, Kamber OS, Kaynak Y, Atakok G, Girit O. Experimental investigation of plastic injection molding: Assessment of the effects of cavity pressure and mold temperature on the quality of the final products. Materials and Design. 2009; 30(8): 3217-3224.

[8] https://www.moldex3d.com/

[9] Arlot S, Celisse A. A survey of cross-validation procedures for model selection. Statistics surveys. 2010; 4: 40-79.

[10] Ho TK. Random decision forests. In: Proceedings of 3rd International Conference on Document Analysis and Recognition; 1995 Aug 14-16; Montreal, QC. IEEE; 1995. p.278-282.

[11] Cassisi C and Montalto, P and Aliotta M and Cannata, A and Pulvirenti A. Similarity measures and dimensionality reduction techniques for time series data mining. In: Karahoca A, editor. Advances in Data Mining Knowledge Discovery and Applications. Rijeka (HR): IntechOpen; 2012. p. 71-96.

[12] Sorjamaa A, Hao J, Reyhani N, Ji Y, Lendasse A. Methodology for long-term prediction of time series. Neurocomputing. 2007; 70: 2861-2869.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210136

Iterative Update of a Random Forest Classifier for Diabetic Retinopathy

Jordi PASCUAL-FONTANILLES [a,1], Aida VALLS [a], Antonio MORENO [a] and Pedro ROMERO-AROCA [b]

[a] *ITAKA-Intelligent Technologies for Advanced Knowledge Acquisition*
Dept. d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili
Avda. Paisos Catalans, 26, 43007, Tarragona, Spain

[b] *Servei d'Oftalmologia, Hospital Universitari Sant Joan de Reus, Institut d'Investigació Sanitària Pere Virgili (IISPV), Universitat Rovira i Virgili, Tarragona, Spain*

Abstract. Random Forests are well-known Machine Learning classification mechanisms based on a collection of decision trees. In the last years, they have been applied to assess the risk of diabetic patients to develop Diabetic Retinopathy. The results have been good, despite the unbalance of data between classes and the inherent ambiguity of the problem (patients with similar data may belong to different classes). In this work we propose a new iterative method to update the set of trees in the Random Forest by considering trees generated from the data of the new patients that are visited in the medical centre. With this method, it has been possible to improve the results obtained with standard Random Forests.

Keywords. Random Forest, Dynamic learning models, Health Care, Diabetic Retinopathy.

1. Introduction

Building decision support systems for medical diagnosis is hard. In the last decade it has become common to use classifiers based on Machine Learning for this task. Usually these systems face a trade-off between sensitivity and specificity.

In this paper we address the problem of Diabetic Retinopathy (DR) classification. As a consequence of diabetes, the blood vessels of the eye may break and generate small blood spots, hemorrhages and exudates. These lesions produce vision loss and may even cause blindness if they are not detected and treated at an early stage. The risk of developing DR can be calculated from some clinical data of the patient, including blood analysis results. Our goal is to improve a DR classification model based on a Fuzzy Random Forest used in the Retiprogram system [1][2]. This model is being tested by a group of ophthalmologists at Hospital Sant Joan de Reus. The general results are good (with a sensitivity

[1] Corresponding Author: Jordi Pascual-Fontanilles. E-mail: jordi.pascual@urv.cat

and a specificity over 75%), but there are still many miss-classifications. Errors are mainly due to the inherent ambiguity of the training examples (very similar patients can belong to different classes) and to the high unbalance between both classes (more than 90% of diabetic patients do not develop DR).

We propose a novel method to take advantage of the data of the new patients which are treated at the hospital. When enough data is gathered, these new examples will be used to perform an update on the random forest model, with the aim of improving its performance. The method also reuses the training examples that the random forest is not able to classify correctly. Even though it has been tested on a fuzzy random forest, the method is suited to be used with standard random forests.

To obtain some experimental results of the proposed method, we used an extended version of the original dataset used in [2]. Data is split in different sets to simulate the arrival of new data to the hospital. A weighted balanced accuracy is used as the performance metric. Several tests have been performed to check the performance of the model during the multiple iterations of the updating method. A study of the best parameters has also been made.

The rest of the paper is organized as follows. Section 2 presents other approaches used to update random forests, consisting on adding weights to the decision trees, or on building trees dynamically from streaming data. In Section 3, we introduce the proposed method for iterative update of the set of trees. In Section 4, we present the dataset and we discuss the obtained experimental results. Finally, Section 5 presents the conclusions and the future work.

2. Related work

The dynamic improvement of classification models based on Random Forests has been studied in the literature. Two different approaches can be found: adding weights to the RF or building online RFs.

Concerning the former approach, weights may be added in four different ways. The first one consists of adding weights at the last step of the RF classifier, when a voting procedure is made to find the majority class ([3], [4], [5]). Each tree on the ensemble has a weight which corresponds to its accuracy. The accuracy is obtained by obtaining the performance of the tree on their out-of-bag samples.

There are also some variations of these methods. For instance, Dogan and Birant [6] proposed initializing all the weights to the same value, and reward the best performing trees on the validation set formed by the out-of-bag samples. Zhukov et al. [7] added a pruning step to replace the worst decision tree so the ensemble can handle concept drift. Decision trees also have a sliding window of stored samples. Each time a new sample has to be evaluated, similar samples from the sliding window are used to recompute the weights based on their errors.

The second option consists on weighting the samples. Kim et al. [8] proposed weighting the samples according to the complexity of classifying them correctly. The trees also have weights, which are computed using the ones on the samples already classified. Yang and Yin [9] considered finding the weight of the decision trees as an optimization problem. For a determined number of epochs, both the weights on the samples and the decision trees are updated to optimize the model.

The third possibility, proposed by Zhong et al. [10], assigns weights to the leaves of the decision trees. That is, each of the rules of the ensemble has a weight based on some performance metric. For regression problems, this method obtains better results than just weighting the decision trees. Finally, there are weighting methods that use different weights for each of the possible output classes. For instance, Zhu et al. [11], Livieris et al. [12] and Utkin et al. [13] use this approach to compensate unbalanced datasets.

The second kind of methods for the dynamic improvement of RFs are based on online random forests. In this case, the training data arrives and is processed in an online manner, that is, in streams. Because the new training samples are not available from the beginning, the methods are not focused on improving an existing model, but on adapting the current one.

Incremental decision trees are one type of online learning methods. For instance, Kalles and Morris [14] and Utgoff et al. [15] proposed variants of ID3 which are incremental. Pecori et al. [16] also proposed using Very Fast Decision Trees to train an ensemble of incremental decision trees, using streaming data. Saffari et al. [17] combined online bagging techniques with extremely randomized forests to build an online random forest. The trees on this random forest grow when new data is fed into the model. A new branch on the tree is created when enough samples are on a node, and are good enough to classify new samples. Similar proposals of online random forests include Mondrian Forests [18] and Adaptive Random Forests [19]. They are also based on growing the trees with the arrival of new training samples. Some incremental approaches also drop some members in the ensemble, and create new ones. Although their objective is to handle concept drifts, and their focus is on processing streams of data.

These two approaches to the dynamic construction of random forests are very different. On the one hand, the weighting methods are applied during the training of the model. They do not need new data because they use the out-of-bag training samples. These approaches are able to improve the results of a standard random forest. Updating of the weights using new data is rarely performed as the effect of a few new incoming samples may not be appropriate. Moreover, the core of the model is not changed, that is, no new rules are learnt.

On the other hand, the main drawback of online random forests is that they need more data than standard random forests to achieve a similar performance. They are not designed to update and improve an existing model, but to construct it in time. They are well suited for applications which have to process continuous streams of data.

3. Proposed method

This paper proposes a novel approach. Instead of modifying the weights of the components of the model or to incrementally build the trees, we propose to change the set of trees that compose the random forest model. This change will be done after collecting a sufficiently large set of new cases that can be used as examples for improving the model. Therefore, it can not be considered a method for streaming data, as some of the works described in the previous section. The proposed architecture is illustrated in Fig. 1. It is composed of three steps.

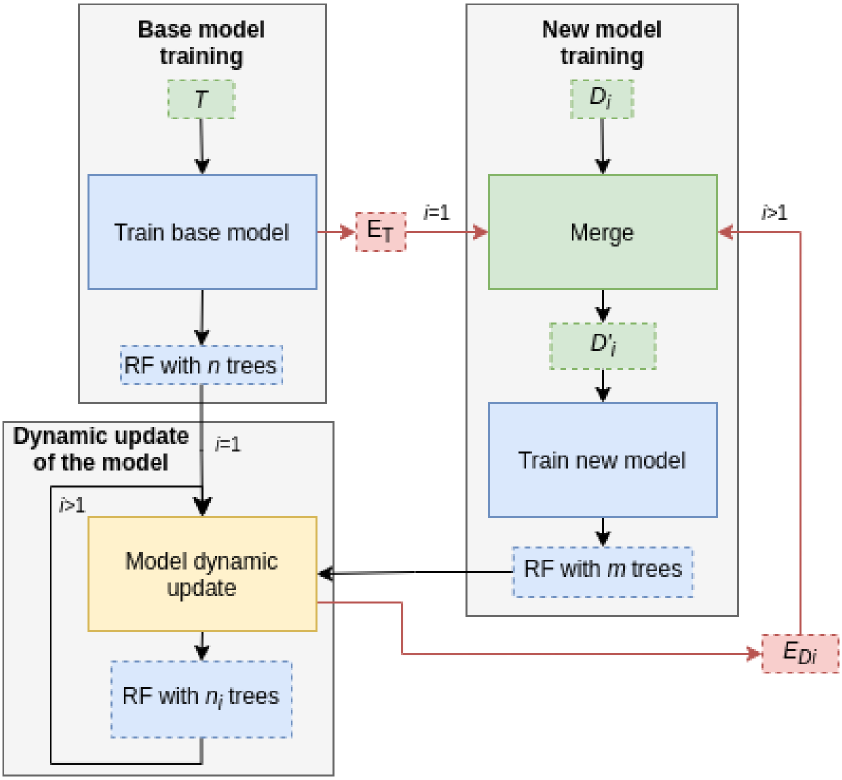


Figure 1. Architecture of the iterative learning of Random forests

1. **Base model training:** The first stage consists on training the base model with a large training dataset, T, which contains labelled examples. With a learning algorithm for Random Forests, we obtain n decision trees, where n is a large number, usually more than 100. During the construction process, the out-of-bag samples are used to compute two metrics for each tree, the specificity and the sensitivity, which are stored on each of the trees. The obtained classification model should be validated with a testing dataset in order to ensure its good performance (this stage is not shown in Fig. 1). Optionally, after creating the base random forest, the training dataset can be used for testing and the samples that are not correctly classified are stored in a file E_T.
2. **New model training:** Every time enough new patients data D_i have been gathered, a new training iteration i is performed. First, the dataset D'_i is generated. For the first training iteration $i = 1$, optionally, the D_i samples and the E_T errors can be merged in a single dataset. With the D'_i samples we train a new random forest but with a low number of trees, because the size of the new training set is small. We get m new decision trees, with $m <<< n$. Their out-of-bag samples are also used to compute the aforementioned metrics for each of these new trees.
3. **Dynamic update:** The base model is updated in this step.

The m new decision trees trained from the D'_i samples are added to the base model. From the $n + m$ trees of the new model, the worst ones are removed. The number of decision trees to remove is fixed to a certain percentage p. The quality metric used to sort the trees is the balanced accuracy, which is defined as an average between specificity and sensitivity, with a weighting factor α.

$$BA = \alpha \cdot sensitivity + (1 - \alpha) \cdot specificity \tag{1}$$

The resulting random forest with n_i trees is taken as the new model to be used by the clinicians until a new set of cases is available, and a new iteration starts. Optionally, the errors of the updated random forest model on the D'_i dataset may be also retrieved and stored in E_{Di}. If so, in the next iteration, the new samples D_i are merged with those error cases E_{Di} in order to enlarge the training dataset of the subsequent iteration.

The use of the sets of wrongly classified examples E_x is optional. The merging of these error examples with the new ones has two purposes. On the one hand, to increase the size of the training set D'_i and, on the other hand, to show again this wrongly classified cases to the learning model in order to be able to build new rules that cover them appropriately. In that way, the model is learning from the past errors. In the next section, the effect of using errors will be studied.

4. Experiments

The diabetic retinopathy risk detection problem has been used to test the proposed iterative method for updating a Random Forest. It is a binary classification problem with two labels. DR=1 means a high risk of suffering from diabetic retinopathy (i.e. positive class), whereas DR=0 means a low risk (i.e. negative class). The experiments have been performed using real data from a total of 25912 diabetic patients. This data includes 9 different attributes, 6 numerical and 3 categorical. The target attribute is the label of the class DR=0 or DR=1.

The data is split in three different datasets: training, validation and testing. The training dataset, T, is used to train the base RF model. It is used to create the model with the largest number of trees (100 trees), hence, it is the dataset with more samples. The validation set is used to simulate the new data that would arrive from the diabetic patients from time to time. We split the validation set in chunks of 800 samples. Each of them is used in a different iteration i during the dynamic updating process. From each of this smaller new training data sets, D_i, the system generates 20 new trees. Then, the dynamic updating stage is done, obtaining the Random Forest model RF_i. Finally, the testing set is used after each iteration to check the performance of the new random forest RF_i. Note that the samples from this testing dataset are not included in the error sets, thus, the model is never trained using these samples.

Table 1 shows the splitting of the data among the three datasets. It can be seen that the datasets are highly unbalanced towards patients without diabetic retinopathy.

Table 1. Diabetic retinopathy patients data

Dataset	Training	Validation	Testing	Total
DR=0 samples	12885 (90%)	6379 (80%)	2514 (70%)	21778
DR=1 samples	1431 (10%)	1621 (20%)	1082 (30%)	4134
Total samples	14316	8000	3596	25912

To obtain the experimental results, we used the aforementioned datasets to build a fuzzy random forest. In this case, the rules use fuzzy variables, which have been defined from the numerical attributes [1]. The fact that the rules are fuzzy does not introduce any change in the methodology for updating the RF model. The training algorithm for fuzzy RF is explained in [2].

The aim of the DR classifier is to improve the detection of patients with high risk of developing diabetic retinopathy, that is, improve the sensitivity of the random forest. For this particular application, it is preferred to misclassify non-DR patients as having the disease (FP), than the other way around (FN). This is due to the very bad consequences of not detecting DR on time, which produces a degradation of the vision that may even cause total blindness.

The method has two parameters: the percentage of trees changed at each iteration, p, and the balancing factor in the calculation of the accuracy, α. We have tested different values of p from 5% to 15%. After some experiments, the percentage was fixed to $p = 10\%$. A lower p value did not produce significant changes on the random forest model. With a higher p, the random forest suffered too many changes and was highly unstable. To determine the α value, diverse tests were performed with α from 0.5 to 0.7. Values higher than 0.5 were selected in order to prioritize the sensitivity over the specificity. We have also compared the performance of the updating methodology when using the misclassified patients (errors in the training datasets) and without using those patients' data (no errors).

The obtained results are presented in Fig. 2, Fig. 3 and Fig. 4. Note that the errors method execution performs one iteration more than the no errors one. This is due to the additional samples obtained from the errors, which permit to create one more training set.

In these 3 figures, we can see similar behaviour on the errors and no errors methods when changing α. The method that does not use errors slightly increases all the performance metrics except for the sensitivity, which slightly decreases. Even though this behaviour could be desired in some applications, in our case, it is critical to improve the sensitivity, as said before.

In contrast, the method that uses the errors data is able to significantly increase the sensitivity. The specificity decreases in this case, but, taking into account the high unbalance between classes, in proportion it does not decrease as much as sensitivity increases. Moreover, the accuracy of the model remains close to 0.75 during all the iterations. For this reason, the approach including data from miss-classified patients is the best suited for this application.

Comparing the three results, the value $\alpha = 2/3$ gives us the best performance. With $\alpha = 0.6$, the sensitivity is a bit unstable and takes more time to increase up to 0.8, compared with using $\alpha = 2/3$. If more weight is given to sensitivity with $\alpha = 0.7$, then specificity decreases too much. So a weight of 2/3 for sensitivity and 1/3 for specificity seems a good trade-off for this application.

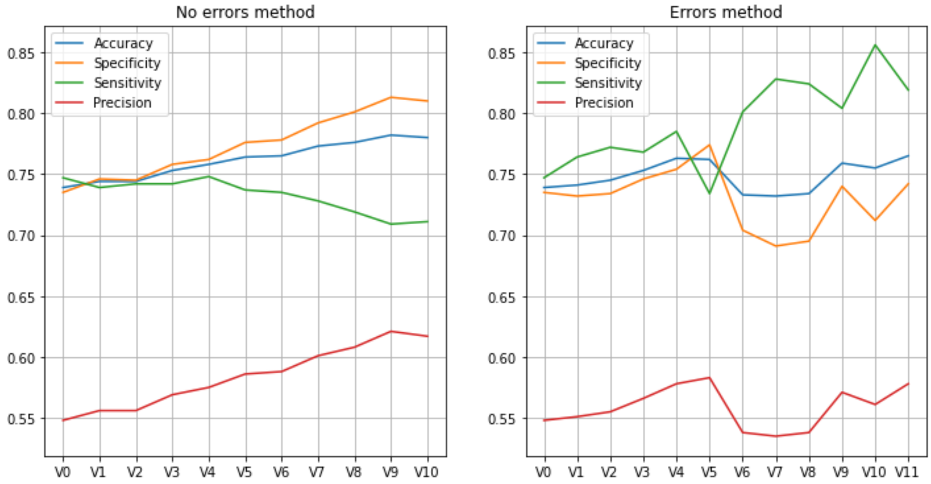


Figure 2. Evolution of the metrics. $p = 10\%$, $\alpha = 0.6$

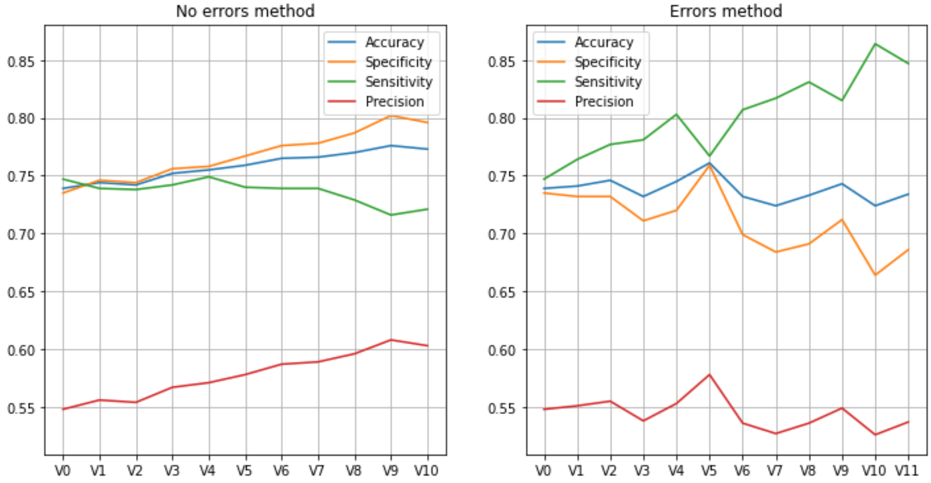


Figure 3. Evolution of the metrics. $p = 10\%$, $\alpha = 2/3$

The behaviour of the selected parameters ($p = 10\%$ and $\alpha = 2/3$) has been further analysed. Fig. 5 shows the number of new generated trees incorporated to the random forest at each iteration. After a few iterations in which the number is quite high, it decreases. It can be seen that the decrease is faster when using the errors data. Reusing the samples in which the random forest is making wrong classifications contributes to a faster learning of a better model.

In Fig. 6, we represent the values in the confusion matrix. In the no errors method, the random forest is able to improve the true negatives and false positives detection rate. Besides, the true positives and false negatives detection rate slightly decrease. This evolution was expected after analysing Fig. 3. In this case, the updated model is more biased towards classifying in class 0 rather than class

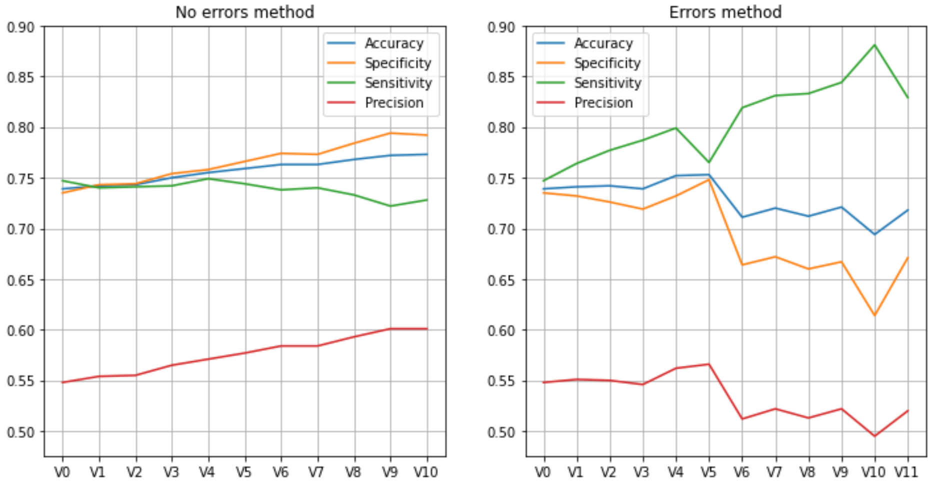


Figure 4. Evolution of the metrics. $p = 10\%$, $\alpha = 0.7$

1. For this reason, it improves the results on detecting the patients without DR (specificity), but slightly decreases the detection of DR patients (sensitivity). In the errors method case, the true positives and false negatives are the ones which are improved on the updated random forest. Even though the negative class detection decreases, considering the high class unbalance, this loss in the negative class is not as important as the gains obtained in the positive class.

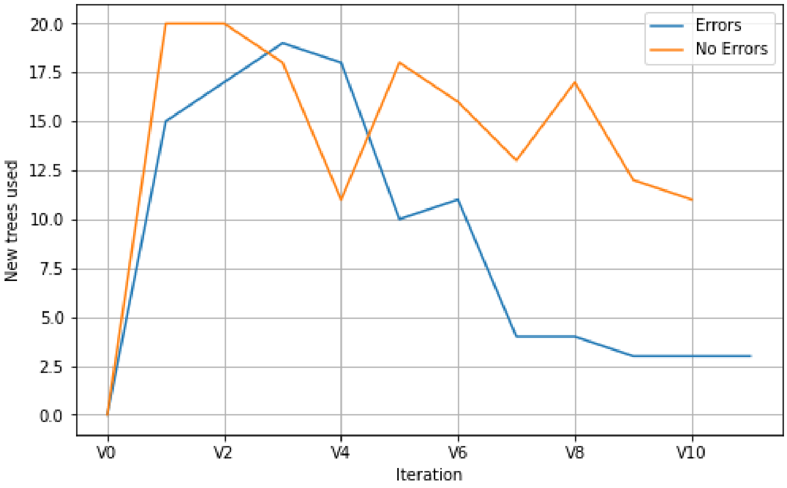


Figure 5. Evolution of new trees used. $p = 10\%$, $\alpha = 2/3$

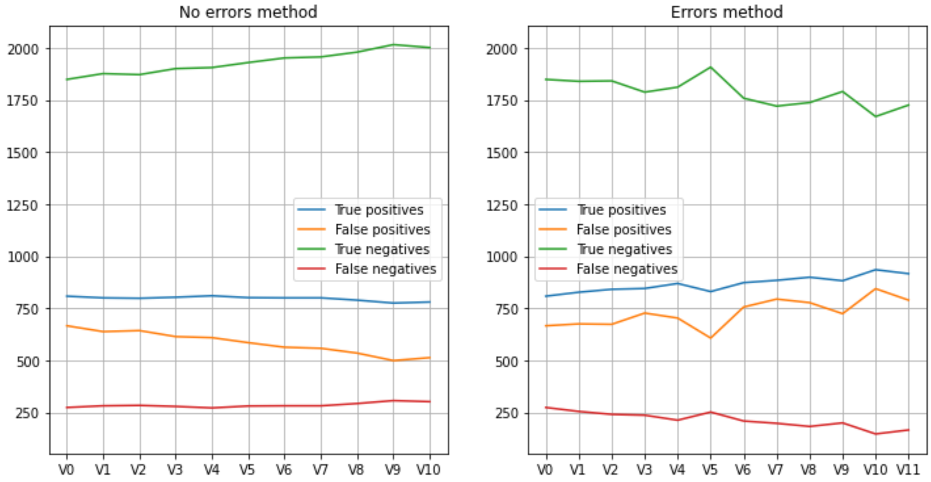


Figure 6. Evolution of the confusion matrix. $p = 10\%$, $\alpha = 2/3$

5. Conclusions and future work

A clinical decision support system for the assessment of diabetic retinopathy risk can help to avoid unnecessary screenings on patients and to reduce the workload of the ophthalmologists. The available resources can also be distributed among the patients, focusing on the ones that really need them.

The method presented in the paper is able to update a random forest in an iterative manner. Given an ill patient, a false negative is more dangerous than a false positive, because the patient would be considered healthy and would not receive the appropriate treatment. Thus, from the two variations of the proposed method, the one using miss-classified samples is the preferred for our use case. This method has been proven to improve the detection of DR=1 patients on a dataset with real data. Even though the detection of DR=0 patients does not improve, given the high unbalance between classes and the inherent ambiguity of this problem, the method seems suitable to improve a random forest model when new data from the patients are available.

As future work, we plan to test the proposed method on other domains. We want to study the possibility of adding weights to the trees. We would like to consider the update of weights of the random forest during the updating process of the model, taking advantage of the new data collected at each iteration. We would also want to study the modifications done to the ensemble, and how they affect the results.

Acknowledgements

This work has been funded by the research project PI18/00169 from Instituto de Salud Carlos III & FEDER funds. The University Rovira i Virgili also supports this work with project 2019PFR-B2-61. The first author has a pre-doctoral FI grant (2021 FI_B 00139) from Secretaria d'Universitats i Recerca de la Generalitat de Catalunya i del Fons Social Europeu.

References

[1] Romero-Aroca P, Valls A, Moreno A, Sagarra-Alamo R, Basora-Gallisa J, Saleh E, et al. A Clinical Decision Support System for Diabetic Retinopathy Screening: Creating a Clinical Support Application. Telemedicine and e-Health. 2019 Jan 1;25(1):31-40.

[2] Saleh E, Valls A, Moreno A, Romero-Aroca P, Torra V, Bustince H. Learning Fuzzy Measures for Aggregation in Fuzzy Rule-Based Models. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer Verlag; 2018. p. 114-27.

[3] Winham SJ, Freimuth RR, Biernacka JM. A weighted random forests approach to improve predictive performance. Statistical Analysis and Data Mining: The ASA Data Science Journal. 2013 Dec 1;6(6):496-505.

[4] El Habib Daho M, Settouti N, Lazouni MEA, Chikh MEA. Weighted vote for trees aggregation in Random Forest. In: International Conference on Multimedia Computing and Systems -Proceedings. IEEE Computer Society; 2014. p. 438-43.

[5] Li HB, Wang W, Ding HW, Dong J. Trees Weighting Random Forest method for classifying high-dimensional noisy data. In: Proceedings - IEEE International Conference on E-Business Engineering, ICEBE 2010. 2010. p. 160-3.

[6] Dogan A, Birant D. A Weighted Majority Voting Ensemble Approach for Classification. In: UBMK 2019 - Proceedings, 4th International Conference on Computer Science and Engineering. Institute of Electrical and Electronics Engineers Inc.; 2019. p. 366-71.

[7] Zhukov A V., Sidorov DN, Foley AM. Random forest based approach for concept drift handling. In: Communications in Computer and Information Science. Springer Verlag; 2017. p. 69-77.

[8] Kim H, Kim H, Moon H, Ahn H. A weight-adjusted voting algorithm for ensembles of classifiers. Journal of the Korean Statistical Society. 2011 Dec 27 ;40(4):437-49.

[9] Yang C, Yin XC. Diversity-Based Random Forests with Sample Weight Learning. Cognitive Computation. 2019 Oct 1;11(5):685-96.

[10] Zhong Y, Yang H, Zhang Y, Li P. Online random forests regression with memories. Knowledge-Based Systems. 2020 Aug 9;201-202:106058.

[11] Zhu M, Xia J, Jin X, Yan M, Cai G, Yan J, et al. Class weights random forest algorithm for processing class imbalanced medical data. IEEE Access. 2018 Jan 3;6:4641-52.

[12] Livieris IE, Kanavos A, Tampakas V, Pintelas P. A weighted voting ensemble self-labeled algorithm for the detection of lung abnormalities from X-rays. Algorithms. 2019 Mar 16;12(3):64.

[13] Utkin L V., Kovalev MS, Meldo AA. A deep forest classifier with weights of class probability distribution subsets. Knowledge-Based Systems. 2019 Jun 1;173:15-27.

[14] Kalles D, Morris T. Efficient incremental induction of decision trees. Machine Learning. 1996 Sep;24(3):231-42.

[15] Utgoff PE, Berkman NC, Clouse JA. Decision Tree Induction Based on Efficient Tree Restructuring. Machine Learning. 1997;29(1):5-44.

[16] Pecori R, Ducange P, Marcelloni F. Incremental learning of fuzzy decision trees for streaming data classification. In: Proceedings of the 11th Conference of the European Society for Fuzzy Logic and Technology, EUSFLAT 2019. Atlantis Press; 2020. p. 748-55.

[17] Saffari A, Leistner C, Santner J, Godec M, Bischof H. On-line random forests. In: 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops 2009. 2009. p. 1393-400.

[18] Lakshminarayanan B, Roy DM, Teh YW. Mondrian Forests: Efficient Online Random Forests. Vol. 27, Advances in Neural Information Processing Systems. 2014.

[19] Gomes HM, Bifet A, Read J, Barddal JP, Enembreck F, Pfharinger B, et al. Adaptive random forests for evolving data stream classification. Machine Learning. 2017 Oct 1;106(910):1469-95.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210137

Anomaly Detection for Diagnosing Failures in a Centrifugal Compressor Train

Ian PALACÍN [a] Daniel GIBERT [a] Jordi PLANES [a,1], Simone ARENA [b], Pier Francesco ORRÙ [b], Maurizio MELIS [c], and Marco ANNIS [c]

[a] *Universitat de Lleida*
[b] *Università degli Studi di Cagliari*
[c] *SARLUX srl*

Abstract. Predicting machine failures is of the utmost importance in industrial systems as it can turn expensive crashes and repair costs into affordable maintenance costs. To this end, this paper presents preliminary work for detecting failures in a centrifugal compressor train based on sensorial data. We show the detection capabilities of a two-step process consisting of: (1) a preprocessing step to reduce the dimensionality of the input data using Principal Component Analysis, and (2) an anomaly detection step using the Mahalanobis distance to detect anomalous observations on the sensors' data. The experiments using real-world data demonstrate the feasibility of our approach and the ability of the method to detect the failures eight days in advance.

Keywords. failure detection, centrifugal compressor, feature reduction, anomaly detection

1. Introduction

During the lifetime of any industrial machine, its components might eventually break, producing an expensive machine breakdown [3]. Thus, diagnosing and detecting machine failures before they occur is essential to avoid the possibility of catastrophic failures, save costs, and keep the machinery running for longer periods of time. Traditionally, the diagnosis of failures in industrial machinery relied on human-expert criteria for the development of an expert system, modeling the expertise knowledge of an expert or group of experts. However, such systems may fail for various reasons including the unavailability of experts and the difficulty to translate the expert knowledge into a concrete method, among others [1].

This increasing complexity of modern machinery aggravates the quality of diagnosis, forcing a shift from a model-based approach to a data-based approach for equipment monitoring and diagnose of failures [11]. As a result, collecting operating and sensor information, from a machine while it is working, has become of the utmost importance

[1] Correspondence to: J. Planes. INSPIRES Research Centre, University of Lleida. C/Jaume II, 69. Lleida, Spain. Tel.: +34 973702764; E-mail: jordi.planes@udl.cat.

to detect anomalies that might manifest a future component failure. Various approaches have been presented in the literature [4,9,6,10] that employ different signal sources to detect failures of industrial machinery, including vibration signals, acoustic emission signals, temperature, and electrical parameters, among others. For instance, Z. Hai-Yang et al. [4] presented a method for diagnosing failures in a reciprocating compressor based on features extracted from vibration signals using Local Mean Decomposition (LMD) and Multi-Scale Entropy (MSE) methods. Y. Wang et al. [9] used acoustic emission signals coupled with the simulated valve motion to diagnose faults in reciprocating compressor valves. In addition, due to the strong correlation and high dimensionality of the data obtained from the sensors installed in complex machinery, dimensionality reduction techniques such as Principal Components Analysis (PCA), have been applied in the literature in order to reduce the number of input variables used to model the data [6,10].

In this paper, a two-step approach to detect failures in a centrifugal compressor train is presented, which relies on the data provided from its sensors. The first step involves the preprocessing of the input data using Principal Component Analysis to reduce its dimensionality. Afterwards, outliers are detected using the Mahalanobis distance in the second step. To evaluate our approach, historical data (throughout 5 years) of the centrifugal compressor train has been used, including temperature, pressure, and vibration of the system valves.

The rest of the paper is organized as follows. Section 2 details the two-step approach presented in this paper. Section 3 presents various experiments used to validate the feasibility of our approach. Lastly, Section 4 presents the final remarks of the paper.

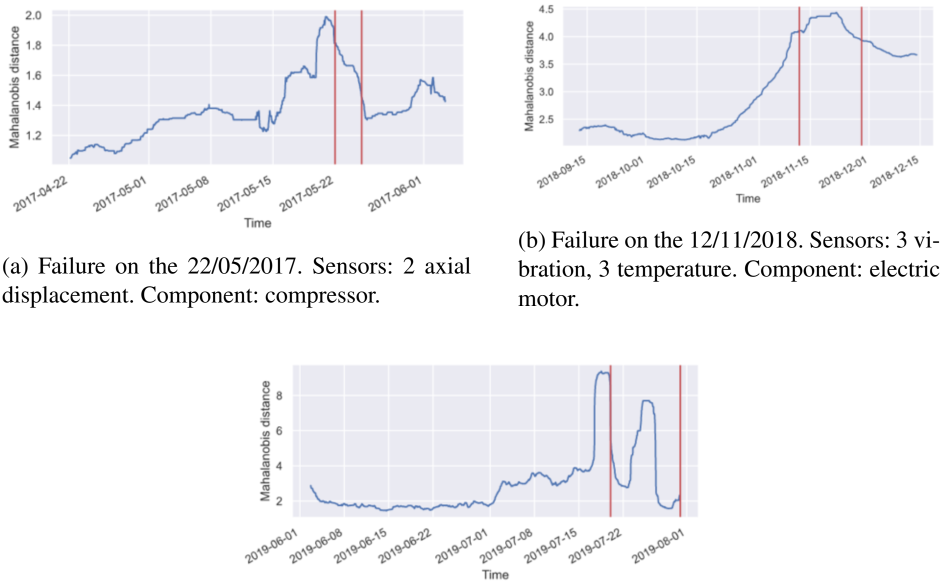
2. Methodology

In this section, we present a two-step process to detect failures in a centrifugal compressor train. The information is provided by 45 sensors distributed along the three main components of the centrifugal compressor train, i.e. the electric motor, the gearbox, and the compressor. The sensors are of five different types: 13 for vibration, 3 for axial displacement, 25 for temperature, 4 for pressure, and 2 for flow. Each sensor may have a different time granularity. The two-step process is as follows:

1. Principal Component Analysis (PCA) is applied to the sensor data in order to reduce the dimensionality of the strongly correlated input variables [7]. PCA provides an orthogonal projection of the data into a lower dimensional linear space, known as the principal subspace, such that the variance of the projected data is maximized [2]. The method allows to select a small number of principal components of the projection. When applied to sensors with different granularity, the coarsest granularity has been chosen.
2. Mahalanobis distance [8] is the multivariate form of the distance measured in units of standard deviation. As outliers are observations that deviate from the global behavior of the majority of data, the usage of the Mahalanobis distance allows to quantify the closeness between the current observation and their distribution and thus, detect when the behavior of the system is anomalous.

3. Experimentation

To evaluate the viability of our two-step process for detecting failures, historical data of a multistage centrifugal compressor for oil and gas service has been used. This data consists of temperature, pressure and vibration of the system monitored with the sensors through five years. In all, ten failures of the compressor were provided. We present three experiments (cf. Figures 1a, 1b and 1c) to visualize our two-step process.



(a) Failure on the 22/05/2017. Sensors: 2 axial displacement. Component: compressor.

(b) Failure on the 12/11/2018. Sensors: 3 vibration, 3 temperature. Component: electric motor.

(c) Failure on the 20/07/2019. Sensors: 3 vibration, 3 temperature. Component: electric motor.

Figure 1. Mahalanobis distance for several sensors in the compressor.

The figures represent the Mahalanobis distance of the data taken at different periods of time, and the red vertical bars represent the periods were failures were reported: Figure 1a shows a failure with a valve in the compressor from 22/05/2017 to 25/05/2017; Figure 1b shows a failure with a valve that started on 12/11/2018 and ended on 29/11/2018; and Figure 1c shows a failure occurred with an oil filter from 20/07/2019 to 31/07/2019. The initial features used to detect the different failures may vary between one failure and another, and were selected using expert knowledge of the system, i.e. some features are more useful than others to detect some of the failures. For instance, the data in Figure 1a comes from the axial displacement of the compressor, whereas Figures 1b and 1c use temperature and vibration information of the electric motor of the compressor train, respectively.

Before the time period of any failure, it can be observed that the Mahalanobis distance dramatically increases, reaching its peak in the failure. This could be due to the degradation of some components in the machine, what may cause the failure. A machine component usually enters into an abnormal state before its failure [5]. In most cases the anomalies were noticeable 8 to 10 days before the failure, providing a good indicator of the feasibility of the Mahalanobis distance to predict failures.

4. Conclusion and Future Work

In order to diagnose failures in a compressor, we have analyzed a two-step process consisting of a feature preprocessing step and an outlier detection step. The process has been evaluated with real-world data, consisting of five years of historical data of a centrifugal compressor train. Results show that our approach may be used to predict failures in industrial systems. Particularly in our case, it can detect anomalies eight days in advance.

A future line of research would be the application of neural networks autoencoders for the anomaly detection in order to automatically trigger the corresponding warnings when the anomalies were detected.

Acknowledgments This work was partially funded by Spanish Project PID2019-111544GB-C22, by the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement 723596 and Grant Agreement 768824, and by 2017 SGR 1537.

References

[1] M. Z. Bell. Why expert systems fail. *Journal of the Operational Research Society*, 36(613619), 1985.

[2] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

[3] H. Groenevelt, L. Pintelon, and A. Seidmann. Production lot sizing with machine breakdowns. *Management Science*, 38(1):104–123, 1992.

[4] Z. Hai-yang, W. Jin-dong, X. Jun-jie, and G. Yi-qi. A feature extraction method based on lmd and mse and its application for fault diagnosis of reciprocating compressor. *Journal of Vibroengineering*, 17(7):3515–3526, 2015.

[5] D. Jiang and C. Liu. Machine condition classification using deterioration feature extraction and anomaly determination. *IEEE Transactions on Reliability*, 60(1):41–48, 2011.

[6] Y. Jiang, B. He, P. Lv, J. Guo, J. Wan, C. Feng, and F. Yu. Actuator fault diagnosis in autonomous underwater vehicle based on principal component analysis. In *2019 IEEE Underwater Technology (UT)*, pages 1–5, 2019.

[7] I. T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, Berlin; New York, 1986.

[8] Sprnger, editor. *Mahalanobis Distance*, pages 325–326. Springer New York, New York, NY, 2008.

[9] Y. Wang, C. Xue, X. Jia, and X. Peng. Fault diagnosis of reciprocating compressor valve with the method integrating acoustic emission signal and simulated valve motion. *Mechanical Systems and Signal Processing*, 56-57:197–212, 2015.

[10] Y. Yinghua, S. Guoqiang, and S. Xiang. Fault monitoring and classification of rotating machine based on pca and knn. In *2018 Chinese Control And Decision Conference (CCDC)*, pages 1795–1800, 2018.

[11] S. M. Zanoli, G. Astolfi, and L. Barboni. Applications of fault diagnosis techniques for a multishaft centrifugal compressor. In *18th Mediterranean Conference on Control and Automation, MED'10*, pages 64–69, 2010.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210138

Pulse Identification Using SVM

Josep PUYOL-GRUART [a,1], Pere GARCIA CALVÉS [a], Jesús VEGA [a,b],
Maria Teresa CEBALLOS [b], Bea COBO [b], Francisco J. CARRERA [b]

[a] *Artificial Intelligence Research Institute (IIIA-CSIC)*
[b] *Institute of Physics of Cantabria (CSIC-UC)*

Abstract.
This paper is about dealing with a noisy signal containing pulses from a detector of X-ray photons. We are designing algorithms to detect pulses, separate them when overlapping, and measure the energy of photons and separation between pulses. In this paper, we present the detection of pulses using SVM.

Keywords. Signal Processing, Machine Learning, Support Vector Machines

1. Introduction

X-ray astronomical observations are performed from space since the Earth's atmosphere blocks this type of radiation. Telemetry limitations to transmit data from the satellites where the X-ray telescopes are installed makes the selection on board of the valuables parts of the signal mandatory. This processing must be done with limited computational resources. We explore machine learning as CNN [1] and SVM to complement current classical methods to improve processing efficiency. As a case study, we use the conditions of the X-IFU [2] detector onboard the ESA Athena mission [3].

2. Experiments

The inputs of our system are noisy digital signals containing single or double pulses, which are the result of X-ray photons impacting Transition Edge Sensor microcalorimeters, the kind of sensor that will be onboard Athena mission. The samples of the signals are separated $6.4\mu s$ (156250 samples/s), and its amplitude is 16 bits data representing the pulse intensity.

We start experimenting with a set of analytically simulated pulses to adjust the training parameters and obtain a first impression of the problem's difficulty. These parameters will be used in the following experiments using a more accurate simulator. This paper is about detecting if there are one or two overlapping pulses in the signal.

[1] Corresponding Author: Josep Puyol-Gruart, Artificial Intelligence Research Institute (IIIA-CSIC), Campus UAB zona 2, 08193 Bellaterra, Spain. Email: puyol@iiia.csic.es

2.1. First experiments

We use a very simple simulator to do the first experiments. The following equation can model the typical response of a sensor to a stimulus. We generate signals with one or two pulses and experiment with adding different amounts of Gaussian noise to the signal.

$$I(s) = \frac{E}{\tau_1 - \tau_2}(e^{\frac{-1}{\tau_1}(s-s_0)} - e^{\frac{-1}{\tau_2}(s-s_0)})$$

$I(s)$ is the pulse intensity at sample s, with constants: E the energy of the pulse, sample constants τ_1 and τ_2 configure the signal's shape, and s_0 is the sample origin. In these experiments $\tau_1 = 30$ and $\tau_2 = 50$ produce shapes similar to the real ones.

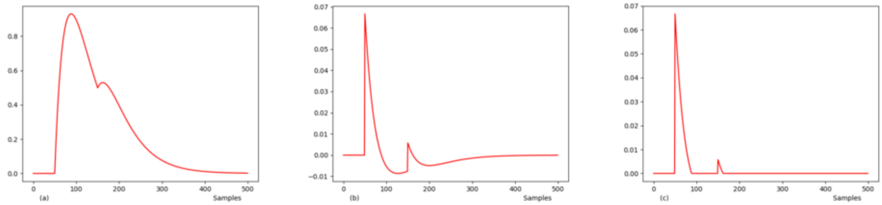


Figure 1. (a) Two pulses starting at samples 50 and 150, with energies 100 and 20 respectively. (b) First derivative (c) First derivative without negative values.

We experimented with the standard signal, its first derivative and cutting the first derivative's negative part (see Figure 1). In all cases, we generate training and test sets of different sizes: 10K, 25K, 50K, 75K and 100K examples with the same number of simple and double pulses. We use 65536 points of amplitude (corresponding to 16 bits of information) and a window of 500 samples. After different experiments with libSVM[4], we use C-SVC, and a kernel with a radial basis function, with $cost = 4$ and the default $\gamma = 1/500$. Rescaling seems not adequate to this problem giving worse results.

In these first experiments, we obtained excellent results, with accuracy greater than 98% with 50K training examples. Never a single pulse is seen as double, but sometimes a double pulse is considered to be single. As we can expect, we obtain the model using the best results with the positive part of the first derivative.

It is essential to notice that preprocessing does not add computational cost to the process. Simple hardware can make this kind of operation before the digitalisation of the signal.

2.2. Experiments with SIXTE simulations

The preliminary experiments described in the section above give us a good idea of the parameters that can be useful to obtain an accurate model with more realistic inputs. Now we will use the same criteria to apply our methodology to a more detailed simulator used for several X-ray missions. SIXTE [5] environment contains the Athena official software for simulations.

We use 60K files containing the examples generated by SIXTE, the same used in the CNN experiments in [1]. We use 45K examples for training and 15K for testing (7500 singles and 7500 doubles). We use the same parameters as in the section before. The input signal is the first derivative without negative elements (*1Der-Cut*).

		Predicted					
		1Der-Cut		*1Der-Cut-Reverse*		*2models*	
		One	Two	One	Two	One	Two
Actual	One	7500	0	7499	1	7499	1
	Two	291	7209	96	7404	40	7460

Table 1. Confusion matrices for the models *1Der-Cut* , *1Der-Cut-Reverse* and the combination of both.

These first experimental results (see Table 1 in column *1Der-Cut*) are similar to those obtained with the first simple simulator. There are no false double pulses, but there are 291 false single pulses (3.88%). All pulses classified by the model as double are true doubles.

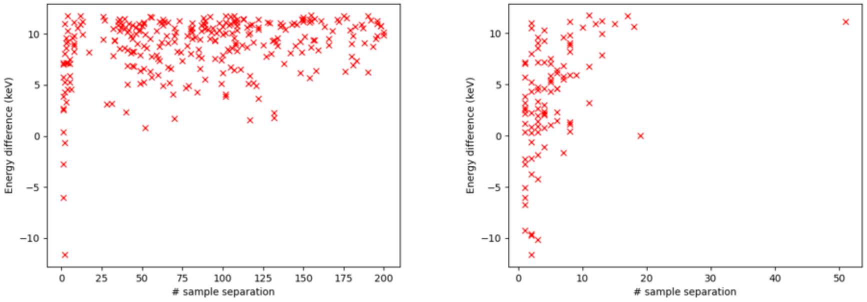


Figure 2. Double pulses detected as singles for the models *1Der-Cut* (left) and *1Der-Cut-Reverse* (right). The X-axis shows the separation of the pulses starting points, and the Y-axis shows their difference of energies.

The separation in time and the difference of energies between two pulses affects the possibility of detecting double pulses. The errors in detecting double pulses are concentrated in pulses separated by relatively few samples or with a significant positive difference of energies, as we can see in Figure 2 left. These errors seem apparent, but we can observe that it is more challenging to detect two pulses when the first pulse has more energy than the second. On the other hand, there are no errors when the first pulse is smaller than the second, except when a few samples separate them. This fact suggests doing a new experiment reading the signals in reverse and obtaining a new model (*1Der-Cut-Reverse*). We convert double pulses with the second smaller to double pulses with the first smaller. The results are in Table 1 in column *1Der-Cut-Reverse* and Figure 2 right. The results are better than in the model from the non reversed signal, with only 96 false single pulses (1.28%).

Finally, we decide to combine the two models (*1Der-Cut* and *1Der-Cut-Reverse*) to deduce if the signal contains single or double pulses. We use the two models and, when both models consider that the pulse is single, then the result is a single pulse. Otherwise, the result is a double pulse.

We can see in Table 1 in column *2models* that the false single pulses are 40 (0.53%) and that now appears one false double pulse. This false double pulse corresponds to a

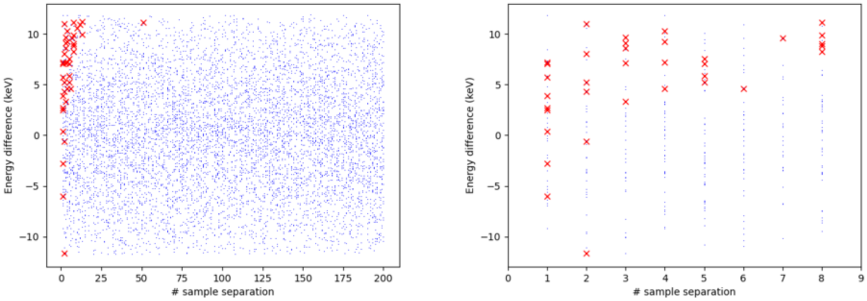


Figure 3. Representation of double pulses detected as singles for the model *2models* (red crosses) and double pulses correctly detected (blue points) At the right, a zoomed version of the left.

single pulse with an energy of 11896eV. It is close to the limit of saturation of the detector (12keV), which can be the reason for the malfunction. The deformation of the signal's shape confuses the model and detects a double pulse.

In Figure 3, we can see at left the distribution of false singles by separation in time and difference of energies. Notice that using the two models, false singles are produced clearly in close pulses or considerable difference of energies, especially when the first pulse is more energetic than the second.

There is an exception in the lonely red cross. This false single pulse corresponds to a double pulse with energies 11319eV and 0.205eV separated by 51 samples. The big difference in energies and the energy of the second pulse (at the limit of detection, 0.2eV) can be the reason for this outlier.

Figure 3 right is a zoom of the left one representing pulses with separations of less than ten samples. We can see that there are nine examples of double pulses only separated by one sample and six by two. When the separation increases, the difference of energies becomes more critical when the first pulse has more energy than the second. The results are very satisfactory, with the model only failing in a small fraction of extreme cases.

3. Conclusions and Future Work

These experiments demonstrate that good detection of single and double pulses is possible using SVM. Although the results are close, an estimate of the computational cost of SVM should be made and compared to CNN [1]. Currently, we are working on estimating the separation between pulses with good prospects. We will also attempt to estimate the pulses' energy, although with less hope since the good results of classical techniques seem challenging to surpass.

References

[1] J. Vega et al., 2020. TES X-ray pulse identification using CNNs. ADASS XXX.
[2] Barret D., et al., 2018, in SPIE. p. 106991G (arXiv:1807.06092),doi:10.1117/12.2312409.
[3] Nandra K., et al., 2013, arXiv e-prints, p. arXiv:1306.2307.
[4] C.-C. Chang and C.-J. Lin, 2011. LIBSVM: A library for support vector machines. TIST, 2(3):127.
[5] Dauser et al. 2019, A&A, 630, A66.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210139

Validation on Real Data of an Extended Embryo-Uterine Probabilistic Graphical Model for Embryo Selection

Adrián TORRES-MARTÍN [a,b], Jerónimo HERNÁNDEZ-GONZÁLEZ [b] and Jesus CERQUIDES [c]

[a] *DEIC, Universitat Autònoma de Barcelona, Bellaterra, Spain*
[b] *Dpt. de Matemàtiques i Informàtica, Universitat de Barcelona, Barcelona, Spain*
[c] *IIIA-CSIC, Campus UAB, Bellaterra, Spain*

Abstract. Embryo selection is a critical step in assisted reproduction (ART): a good selection criteria is expected to increase the probability of inducing pregnancy. In the past, machine learning methods have been used to predict implantation and to rank the most promising embryos. Here, we study the use of a probabilistic graphical model that assumes independence between embryos' individual features and cycles characteristics. It also accounts for a third source of uncertainty attributed to unknown factors. We present an empirical validation and analysis of the behavior of the model within real data. The dataset describes 604 consecutive ART cycles carried out at Hospital Donostia (Spain), where embryo selection was performed following the Spanish Association for Reproduction Biology Studies (ASEBIR) protocol, based on morphological features. The performance of our model is evaluated with different metrics and the predicted probability densities are examined to obtain significant insights about the process. Special attention is given to the relation between the model and the ASEBIR protocol. We validate our model by showing that its predictions show correlation with the ASEBIR score when the score is not provided as a feature. However, once the selection based on this protocol has taken place, our model is unable to separate implanted and failed embryos when only embryo individual features are used. From here, we can conclude that ASEBIR score provides a good summary of morphological features.

Keywords. Assisted Reproductive Technologies, Embryo Selection, Machine Learning, Probabilistic Graphical Models, Learning from Label Proportions

1. Introduction

Assisted reproductive technologies (ARTs) are a set of invasive medical techniques that attempt to induce a pregnancy, used mainly to address infertility. Each trial of a reproduction treatment applying a suitable ART is known as a cycle. When a woman undergoes a cycle, she follows a treatment of ovarian stimulation for several weeks. Then, oocytes are retrieved and fertilized, and the resulting embryos are cultured for several days. Afterwards, the most viable embryos are selected to transfer to the uterus. After transference, the occurrence of embryo implantation determines the process of the cycle. However, for a transfer, current techniques are able to determine the number of embryos that implanted, but unable to identify individually which ones implanted. The probability of pregnancy could be increased by transferring a larger number of embryos [1], but this leads to higher multiple-birth rates, which is considered risky for both mother and the developing fetuses [1, 2]. In fact, in many countries there are legal restrictions limiting the number of embryos transferred (e.g., Spanish law limits it to 3). Therefore, the selection of the most viable embryos is a critical step to optimize the probability of pregnancy.

Embryo selection is a complex and partially subjective task. The evaluation of embryos is based mainly on their morphological features. Initially, the lack of consensus in this assessment made it impossible to compare results across centres [3]. A unified criteria was created to address this problem: the ASEBIR protocol [4]. This method classifies embryos into a categorical scale (A,B,C,D) using morphological criteria. In recent years, machine learning techniques have been used to assist clinicians in embryo selection and pregnancy prediction [5, 6, 7, 8]. Most of them rely on supervised classification, meaning that only the embryos whose outcome is known (all embryos in the cycles were implanted or none were) are used for training. However, novel methods [7] try to benefit from cycles with partial implantation (not all the transferred embryos were implanted).

The model considered in this paper also deals with cycles with partial implantation. Our model accounts for the factors handled by clinicians in their standard practice as possible determinants of the success of an ART cycle. It assumes independence between embryos and cycles and takes into account a third source of uncertainty corresponding to unknown factors. The main goal of this work is to validate the model using real data, and to study the correlation with ASEBIR score. The paper is organized as follows. Next, we describe the dataset. In Section 2, the model is presented as well as the learning algorithm. In Section 3 the experimental setup is explained, introducing the different probabilistic classifiers and metrics. Then, their results are shown and discussed. Finally, in Section 5 conclusions are drawn and a few open lines for future work are presented.

1.1. Data

The database, originally studied in [7], was collected by the Unit of Assisted Reproduction of the Hospital Donostia (Spain) throughout 18 months (January 2013 - July 2014). It contains 604 cycles of an ART treatment and 3125 associated embryos. Each cycle has a certain number of embryos associated (between 1 and 18), only some of which will actually be transferred (between 1 and 3). Cycles are described by 25 features, mainly related to the female patient, the sperm donor and the stimulation procedure. Embryos are described by 20 features, out of which 13 are morphological features.

Out of the 604 cycles, 192 resulted in a pregnancy with 253 embryos implanted. Of these successful cycles, in 57 of them all the embryos were implanted (108 embryos). In total, the outcome of 947 embryos is known (all embryos implanted in a cycle or none), for 307 we have only the label proportions (in cycle with not all embryos implanted), and for the rest, 1871 embryos, we do not have any information (not transferred embryos).

2. Method

In this work we employ a probabilistic model originally presented in [9] that uses the available information from both cycles and individual embryos, and considers a third source of uncertainty related with unknown factors [10].

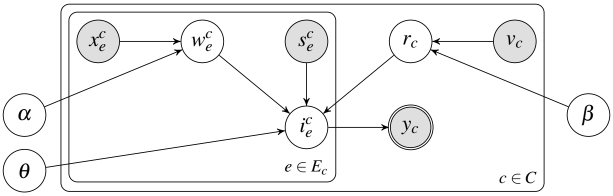


Figure 1. Graphical description of the proposed model. Shadowed nodes represent observed variables. Double line denotes a deterministic variable.

2.1. General probabilistic model

The main assumption of the model is that the probability of an embryo being willing to implant given its own features is independent of the corresponding cycle's features. Similarly, the probability of a cycle being willing to let embryos implant given its own features is independent of the embryos' individual features. Hence embryos and cycles are modeled independently. Moreover, the main novelty is that our model accounts for unknown factors that affect ART success [10] which cannot be explained by the available data. This third source of possible error is included in the model as a Bernoulli distribution with parameter θ_1. The probability of implantation of a high-quality embryo within a cycle willing to let embryos implant is θ_1. If the available information were capable of perfectly predicting the outcome of the process (i.e., no unknown factors), this parameter would be $\theta_1 = 1$. If one of the components (embryo or cycle) is not deemed as good enough to allow implantation then the probability of implantation is directly 0.

Let x_e^c denote the characteristic features of embryo e included in cycle c. Denote by w_e^c a Boolean random variable that represents whether the embryo is willing to implant. This variable w_e^c is modeled by the probability distribution $p(w_e^c \,|\, x_e^c; \alpha)$, where α is the hyperparameter of such distribution. Similarly, let v_c denote the features of cycle c. Denote by r_c a Boolean random variable that represents whether cycle c is willing to let embryos implant, modeled by the probability distribution $p(r_c \,|\, v_c; \beta)$, with hyperparameter β. Both w_e^c and r_c are modeled using probabilistic classifiers.

Let s_e^c denote an observed variable that tells whether embryo e is transferred in cycle c. Denote by i_e^c a Boolean random variable that represents whether embryo e implants in cycle c, modeled by a Bernoulli distribution $i_e^c \sim \text{Bernoulli}(\theta_{w_e^c \cdot r_c \cdot s_e^c})$, given w_e^c, r_c and s_e^c. That is, $\theta_{w_e^c \cdot r_c \cdot s_e^c}$ is only θ_1 when all three variables are positive.

Finally, let y_c denote an observed variable that tells the number of embryos implanted in a cycle. It is just the sum of the i_e^c variables modeling embryo implantation (deterministic), $y_c = \sum_{e \in E_c} i_e^c$, where E_c is the set of embryos associated to cycle c.

Figure 1 shows the complete graphical representation of the model. The shadowed variables are the observed ones (features and final number of implantations per cycle), and θ, α and β are the hyperparameters of the three probability distributions that we are modeling. The other three white nodes w_e^c, i_e^c and r_c represent latent variables, which generally need to be inferred. In some cases the value of y_c is enough to deduce the value of these variables. For example, if $y_c > 0$ then we know that this cycle is willing to let embryos implant ($r_c = 1$). However, if $y_c = 0$ we do not know which was the actual cause of failure: the embryo, the cycle or an unknown factor. The complete joint probability is

$$p(\mathbf{x}, \mathbf{w}, \mathbf{v}, \mathbf{r}, \mathbf{s}, \mathbf{i}, \mathbf{y}; \alpha, \beta, \theta) = p(\mathbf{w}|\mathbf{x}; \alpha) p(\mathbf{x}) p(\mathbf{r}|\mathbf{v}; \beta) p(\mathbf{v}) p(\mathbf{s}) p(\mathbf{y}|\mathbf{i}) p(\mathbf{i}|\mathbf{w}, \mathbf{r}, \mathbf{s}; \theta) \quad (1)$$

The goal of the learning algorithm is to estimate the set of hyperparameters parameters θ, α and β that maximize the conditional probability:

$$p(\mathbf{y}|\mathbf{x},\mathbf{v},\mathbf{s};\alpha,\beta,\theta) = \sum_{\mathbf{r}} p(\mathbf{r}|\mathbf{v};\beta) \sum_{\breve{i}\in\mathbb{I}_{\mathbf{s},\mathbf{y}}} \sum_{\mathbf{w}} p(\breve{\mathbf{i}}|\mathbf{w},\mathbf{r},\mathbf{s};\theta)p(\mathbf{w}|\mathbf{x};\alpha) \tag{2}$$

where $\mathbb{I}_{\mathbf{s},\mathbf{y}}$ is the set of vectors i compatible with the selections $\{s^c_e\}$ and the known outcomes $\{y_c\}$. E.g., in a cycle with 4 embryos, where only the first and third are selected and only one of them was implanted, the possible vectors are $[1,0,0,0]$ and $[0,0,1,0]$.

2.2. EM Algorithm

In the presented model there are latent variables (w^c_e, i^c_e and r_c) whose value is generally unknown. We use an Expectation-Maximization (EM) algorithm [11] to learn in this scenario, combining the completion (expectation) of these latent variables with the estimation of the hyperparameters θ, α and β maximizing the log-likelihood.

For each cycle c we consider a pair of weights $q(r_c = r)$ associated to the two possible values of r_c, $r \in \{0,1\}$. These weights are computed as the likelihood of obtaining $r_c = r$ taking into account the whole model, not just the features of the cycle:

$$q(r_c = r) \propto \Big(\sum_{\mathbf{i}^c\in\mathbb{I}_{\mathbf{s}^c_:,y_c}} \prod_{e} \sum_{w^c_e} p(i^c_e|w^c_e, r_c = r, s^c_e;\theta)p(w^c_e|x^c_e;\alpha)\Big) p(r_c = r|v_c;\beta). \tag{3}$$

Similarly, for each embryo e in the cycle we compute the weights corresponding to the two values of w^c_e, $w \in \{0,1\}$:

$$q(w^c_e = w) \propto \sum_{r_c} \left(\sum_{\mathbf{i}^c\in\mathbb{I}_{\mathbf{s}^c_:,y_c}} p(i^c_e|w, r_c, s^c_e;\theta)p(w|x^c_e;\alpha)\cdot \prod_{e'\neq e} \sum_{w^c_{e'}} p(i^c_{e'}|w^c_{e'}, r_c, s^c_{e'};\theta)p(w^c_{e'}|x^c_{e'};\alpha) \right) p(r_c|v_c;\beta) \tag{4}$$

Finally, the weights associated to each possible combination ($\mathbf{i} \in \mathbb{I}_{\mathbf{s}^c_:,y_c}$) for $\mathbf{i}^c$ are:

$$q(\mathbf{i}^c = \mathbf{i}) \propto \sum_{r_c} \Big(\prod_{e} \sum_{w^c_e} p(i_e|w^c_e, r_c, s^c_e;\theta)p(w^c_e|x^c_e;\alpha)\Big) p(r_c|v_c;\beta) \tag{5}$$

Our algorithm starts with the initialization, where the weights are randomly assigned and normalized (to sum up to 1). If the real value of the variable is known, these values are fixed (e.g., if $y_c > 0$ then $q(r_c = 1) = 1$ and $q(r_c = 0) = 0$). Then, it repeats iteratively:

Expectation: The unfixed weights are updated with Equations 3, 4 and 5, using the current fit of the model $(\hat{\alpha}, \hat{\beta}, \hat{\theta}_1)$.

Maximization: Hyperparameters $(\alpha, \beta, \theta_1)$ are re-estimated. For α and β, we retrain the probabilistic classifiers with the new weights obtained from the previous E-step. For θ_1, the maximum likelihood estimator is:

$$\hat{\theta}_1 = \frac{\sum_c \sum_{\mathbf{i}^{c'} \in \mathbb{I}_{\mathbf{s}_:^c, y_c}} \sum_e q(\mathbf{i}^{c'}) q(r_c = 1) q(w_e^c = 1) i_{e'}^c}{\sum_c \sum_{\mathbf{i}^{c'} \in \mathbb{I}_{\mathbf{s}_:^c, y_c}} \sum_e q(\mathbf{i}^{c'}) q(r_c = 1) q(w_e^c = 1)} \tag{6}$$

The method iterates until the stopping condition is met (maximum number of iterations or convergence of weights). It is run multiple (10) times with different initialization to mitigate the local-maximum problem of EM algorithms.

3. Experimental setup

The main goal of this project is to analyze the performance of the general probabilistic model proposed in Section 2.1 and to compare the performance with different probabilistic classifiers. Moreover, we study the effect in our model of the ASEBIR quality score [4], and whether both our model and this score agree on the embryo selection.

Our model uses probabilistic classifiers to predict the probability that an embryo is willing to implant, $p(\mathbf{w}|\mathbf{x};\alpha)$, and that a cycle is willing to let embryos implant, $p(\mathbf{r}|\mathbf{v};\beta)$. Different classifiers may perform differently depending on the context. In order to make a fair comparison between the various models we use three different probabilistic classifiers: (i) Extra-trees classifier (ETREES) (ii) Gradient Boosting (GBOOST) and (iii) Logistic Regression (LR).

Because of the weakly supervised nature of the problem [12], the evaluation of the models is not trivial and needs to be properly addressed in order to make a fair comparison. For instance, a large fraction of embryos in the dataset were not actually transferred; hence they are not labeled as implanted or not. We use them in our EM learning algorithm but they cannot be used to assess model performance. Moreover, a part of the transferred embryos have no label: when only some of the embryos in their cycle were implanted. However, for these, we do know the proportion of the embryos that were implanted. This information should be used to take full advantage of the data.

Also the interpretation of the predictions needs proper consideration. For instance, the full model gives the probability of implantation of an embryo in a certain cycle assuming independence between embryo and cycle. In fact, we can compute the probability of both embryos and cycles of being appropriate for ART directly with the respective probability classifier. This means that, if we only want to rank a set of embryos according to their *quality*, we could use just the embryo classifier trained within the whole model.

To test the performance of the model and obtain relevant metrics and probability densities, we use 5-fold cross validation. The resulting measures are averaged to obtain a final evaluation metric. Most of the metrics used here need the probability of implantation of an embryo in a cycle, which is given by:

$$p(i_e^c = 1|x_e^c, s_e^c, v_c; \alpha, \beta, \theta) = p(i_e^c = 1|w_e^c = 1, s_e^c, r_c = 1; \theta) p(w_e^c = 1|x_e^c; \alpha) p(r_c = 1|v_c; \beta). \tag{7}$$

where $p(i_e^c = 1|w_e^c = 1, s_e^c, r_c = 1; \theta) = \theta_1 \cdot s_e^c$. Remember that if $s_e^c = 0$, $p(i_e^c = 1|w_e^c, s_e^c = 0, r_c; \theta) = 0$. That is why the evaluation is only performed with embryos which were transferred ($s_e^c = 1$). The other two terms in Eq. 7 are given by the probabilistic classifiers.

Performance is assessed in terms of different metrics. To test the ability to predict embryo implantation, we use only the embryos whose fate is known (i.e., those belonging to completely implanted cycles or failed cycles) and measure the AUC-ROC [13].

Table 1. Metrics and control measures obtained using 5-fold cross validation

Classifier	AUC-ROC	LP-loss	loglikelihood	AUC-ROC	LP-loss	loglikelihood
ETREES	0.64 ± 0.07	$\mathbf{0.54 \pm 0.05}$	1.45 ± 1.59	0.64 ± 0.05	$\mathbf{0.54 \pm 0.05}$	1.27 ± 1.57
GBOOST	$\mathbf{0.71 \pm 0.04}$	0.72 ± 0.03	$\mathbf{0.45 \pm 0.05}$	$\mathbf{0.73 \pm 0.07}$	0.73 ± 0.07	$\mathbf{0.43 \pm 0.06}$
LR	0.63 ± 0.08	0.60 ± 0.05	0.51 ± 0.10	0.62 ± 0.08	0.64 ± 0.07	0.52 ± 0.10
	Full model			Full Model, hidden quality		

To account also for the partially implanted cycles, we use the label proportion loss (LP-loss) and the negative log-likelihood. LP-loss measures how close the real and predicted label proportions are. For each cycle, the difference between the number of embryos predicted as implanted and the actual number of implanted embryos is taken in absolute value. The LP loss is the mean value of these differences. Similarly, we might want to consider how confident is the model in predicting each of these labels. For that matter, we use the negative log-likelihood. As most of the embryos do not have a true label to compare with, we compute this measure cycle by cycle, calculating the likelihood of the real number of implanted embryos within the learnt model. Let N_c be the number of transferred embryos in cycle c, and y_c the number of implanted ones. The negative log-likelihood is

$$\mathscr{L}(\mathbf{Y};\alpha,\beta,\theta) = -\frac{1}{B}\sum_{c}\sum_{j=0}^{N_c} \mathbb{1}[y_c = j]\log p(y_c), \tag{8}$$

where B is the total number of cycles and $p(y_c)$, the probability of cycle c having y_c implanted embryos, is,

$$p(y_c) = \sum_{i^c \in \mathbb{I}_{y_c}} \prod_{e} [i^c_e p(i^c_e = 1) + (1 - i^c_e) p(i^c_c = 0)] \tag{9}$$

where $p(i_c)$ is given by Eq. 7 and $\mathbb{I}_{y_c}$ consists of the possible joint assignment of value (vector) to all the $\{i^c_e\}_{e \in E_c}$, as explained in the context of Eq. 2.

4. Results and discussion

A relevant point is whether our model agrees with the ASEBIR score. In our dataset, we have this score as a feature, as well as all the factors used to compute it. To study the agreement, we trained the model in two different ways: with and without this quality score included as a feature of embryos. In Table 1 we show the metrics obtained for each probabilistic classifier and for both models (with and without ASEBIR score feature). Observe that there are no significant differences between the two different models. The model seems not to be directly using the feature as a discriminant for implantation. It must be gathering that information from the other features in the dataset which are, in fact, the ones used in their protocol [4].

In terms of performance, GBOOST seems to be the best one according to AUC-ROC and negative log-likelihood. ETREES and LR classifiers are both similar regarding AUC-ROC but their log-likelihood values are rather different. A critical difference between these two measures is that they use a different set of embryos for evaluation. AUC-ROC

Table 2. Estimated parameter θ_1 for the three different classifiers.

Model	Classifier	θ_1	Model	Classifier	θ_1
Full Model	ETREES	0.60 ± 0.04	Full Model (Hidden quality)	ETREES	0.58 ± 0.04
	GBOOST	0.49 ± 0.00		GBOOST	0.49 ± 0.01
	LR	0.52 ± 0.01		LR	0.51 ± 0.00

is calculated using only embryos whose outcome is known, whereas log-likelihood uses all transferred embryos, evaluating cycle by cycle the proportion of implanted embryos. Thus, ETREES performs relatively well in a pure classification task (implantation or not), but it fails on estimating the probability of more uncertain cases.

Table 2 shows the mean estimation of the parameter θ_1 obtained with each classifier and model, over the different CV folds. The standard deviation is quite low for all the classifiers, implying a consistent estimation. This parameter is the probability that a good embryo will actually get implanted in a good cycle. It represents the third source of failure for implantation of our model, and accounts for all unknown factors.

For the GBOOST and LR classifiers, the mean value of θ_1 is close to 0.5. This means that these models, even when the classifiers consider that both embryo and cycle are promising, expect that only half of these pairs will succeed. The ETREES classifier estimates a noticeably higher $\theta_1 = 0.58$. This might suggest that this model has a higher confidence on the judgement of its embryo and cycle classifiers. Unfortunately, this confidence does not translate into better results (see Table 1).

To fully grasp the behaviour of the models, we also analyze the different predicted probability densities output by them. Figure 2 displays the densities, separated for successful and failed cycles, of (i) whether the embryo is willing to implant, (ii) whether the cycle is willing to accept embryos, and (iii) whether the ART treatment is leading to a pregnancy (whole model). The ideal classifier would separate clearly the curves of each class for the third case (right column). The results of all classifiers are quite similar: Although intersection between both densities is considerable, the mode of the density for successful treatments (pregnancy) is clearly to the right regarding that of the failed treatments. This means that, on average, *the models predict the actual implanted embryos as more likely to implant than the failed ones*.

In the first column of Figure 2, the probability of deeming an embryo as willing to implant is practically the same for successful and failed treatments. At a first sight, one could think that embryos are not relevant to predict a pregnancy. Nevertheless, it is noteworthy that the embryos employed in this study are only the ones that were transferred. And, transferred embryos are usually the best embryos as selected by the embryologists, that is, all the embryos that we observed were considered as good-quality ones by the specialists. Instead, most of the predictive power seems to come from the cycle. In the middle column of Figure 2, it can be observed that the classifier gives a higher probability of being a cycle willing to be implanted to those treatments that induced a pregnancy. All this could mean that *the protocol followed by the embryologists performs well in selecting the best embryos based on the morphological features*. Our model is not able to further discriminate the embryos based on this data (the same they used) alone.

Figure 3 displays, in a similar way, different probability densities separating on the ASEBIR score of the involved embryo. For this experiment, we hide the ASEBIR score feature from the model. Under the independence hypothesis, the quality of an embryo should not affect the probability that a cycle is in good conditions and, for the most part

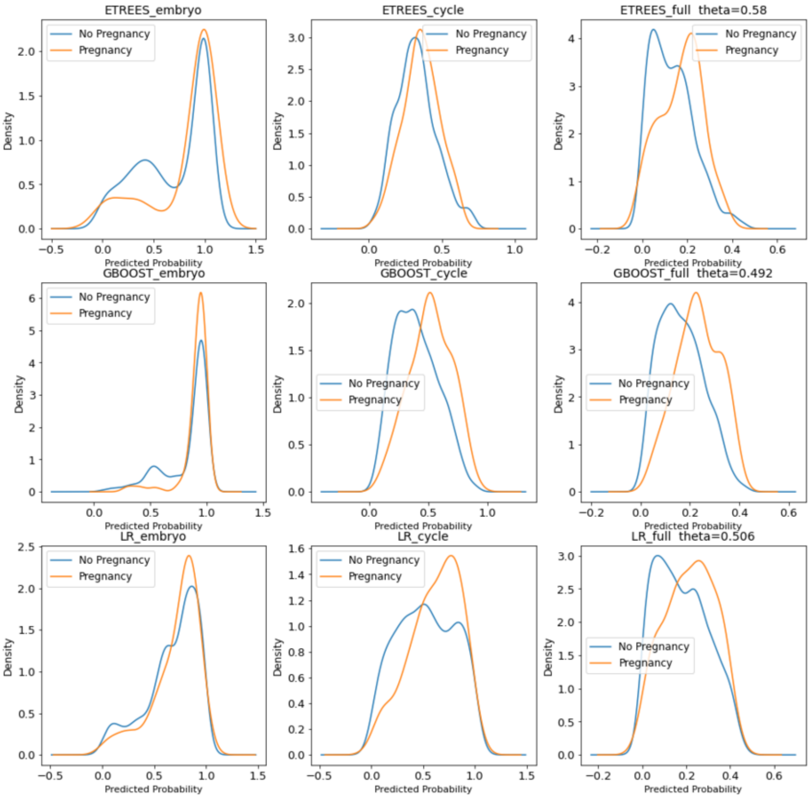


Figure 2. Density of the predicted probabilities for an embryo to be willing to implant, for a cycle to be willing to let embryos implant and for the pair embryo-cycle to actually induce pregnancy. The figure shows the different probability densities depending on the true outcome of each embryo-cycle pair (induce pregnancy or not). Each row corresponds to a different probabilistic classifier (ETREES, GBOOST and LR).

of it, we observe that the embryo information has not leaked into the cycle classifier. However, with ETREES there is a slight disparity in favor of treatments using embryos of good quality. Embryo quality has the highest impact on the probability of considering an embryo as willing to implant. All classifiers separate quite well the best (A) and worst (D) quality embryos. ETREES and GBOOST seem not to differentiate embryos of medium quality (B and C) completely, while LR does separate them slightly. For all classifiers the embryo quality does translate well into the final prediction of implantation. Note that this does not validate the model regarding implantation, but it implies that *the model mostly agrees with the ASEBIR score in the selection of the most promising embryos based on this set of features*.

5. Conclusions

In this paper, we address embryo selection for ARTs using a probabilistic model that assumes independence between embryos and cycles. Using morphological data for each individual embryo and characteristics about the cycle, the model is able to predict im-

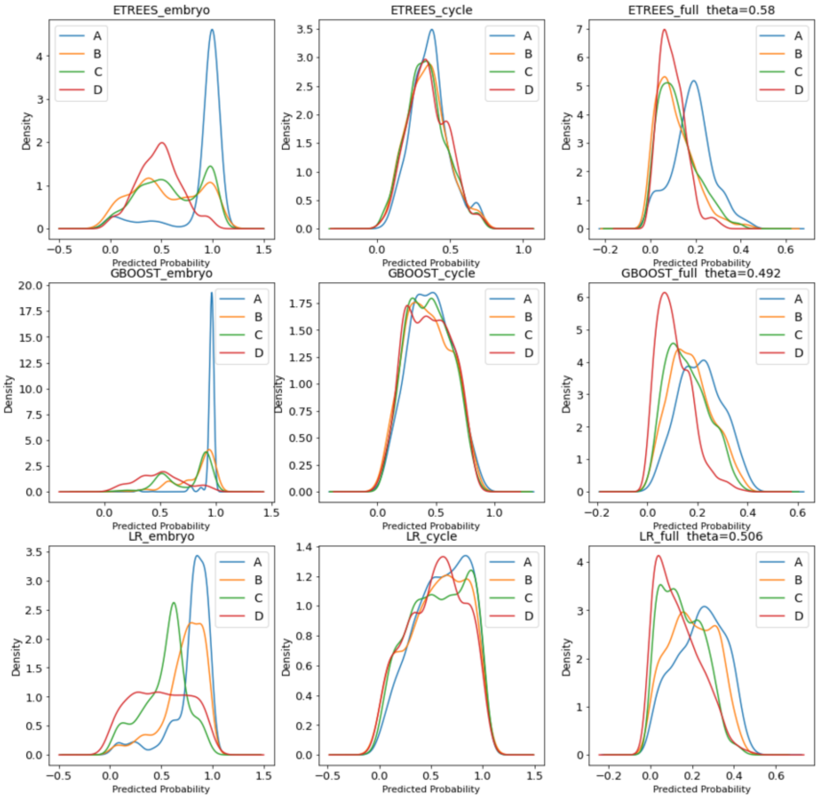


Figure 3. Density of the predicted probabilities for an embryo to be willing to implant, for a cycle to be willing to let embryos implant and for the pair embryo-cycle to actually implant. The figure shows the different probability densities depending on the ASEBIR quality score given to the embryo (A, B, C or D). Each row corresponds to a different probabilistic classifier (ETREES, GBOOST and LR).

plantation. The performance of the model is tested using different classifiers which evaluate the goodness of the embryos and cycles. The Gradient Boosting classifier showed the best results both in terms of AUC-ROC and negative log-likelihood.

The probability densities obtained from the predictions provided helpful insights to understand the behaviour of the model. We studied the effect of the ASEBIR embryo quality score within our model. We have not observed differences between models learnt with and without the ASEBIR score directly as a feature. The probability densities grouped by this quality feature show a clear separation between groups (especially between the best and worst grades), using both models. We have observed that, once embryologists made their selection, the model does not provide more information about individual embryos. This might indicate that the protocol followed by the embryologists is already extracting most of the value out of the morphological data.

There are different research lines open after this exploration of the behaviour of the model in relation to the ASEBIR protocol. We plan to enlarge our experimental setup to obtain a deeper understanding of the intricacies of the model. Another direction would be to conceive new, maybe simpler, models to test the assumptions of our current model (independence between embryos and cycles, awareness of a third source of error, etc.).

Acknowledgments: J.H.-G. is a Serra Húnter Fellow. This work was partially supported by projects Crowd4SDG (No 872944) and Humane-AI-net (No 952026), funded by EU Horizon 2020 program, and project CI-SUSTAIN (PID2019-104156GB-I00), funded by Spanish Ministry of Science and Innovation. We would like to thank Olga Valls for providing the source code of the model, and the physicians from Hospital Donostia (Spain) who collected the data and granted us permission to use it.

References

[1] L Engmann, N Maconochie, S Tan, and J Bekir. Trends in the incidence of births and multiple births and the factors that determine the probability of multiple birth after IVF treatment. *Human Reproduction*, 16:2598–605, 12 2001.

[2] ESHRE Campus Course Report. Prevention of twin pregnancies after IVF/ICSI by single embryo transfer. *Human Reproduction*, 16(4):790–800, 04 2001.

[3] I Cuevas-Siz, M C Pons, M Vargas, A Mendive, N Enedáguila, M Solanes, B Carrasco, J López, A Bonet, and M Acosta. The Embryology Interest Group: updating ASEBIR's morphological scoring system for early embryos, morulae and blastocysts. *Medicina Reproductiva y Embriologa Clnica*, 5, 02 2018.

[4] M. Ardoy and G. Calderon. Clinical embryology papers: Asebir criteria for the morphological evaluation of human oocytes, early embryos and blastocysts. *Asociacin para el Estudio de la Biologa de la Reproduccin (ASEBIR)*, 2008.

[5] G Corani, M C Magli, A Giusti, L Gianaroli, and L M Gambardella. A bayesian network model for predicting pregnancy after in vitro fertilization. *Computers in biology and medicine*, 43:1783–92, 11 2013.

[6] F. Guérif, A. le Gouge, B. Giraudeau, J. Poindron, R. Bidault, O. Gasnier, and D. Royère. Limited value of morphological assessment at days 1 and 2 to predict blastocyst development potential: a prospective study based on 4042 embryos. *Human reproduction*, 22 7:1973–81, 2007.

[7] J Hernández-González, I Inza, L Crisol-Ortíz, M A Guembe, M J Iñarra, and J A Lozano. Fitting the data from embryo implantation prediction: Learning from label proportions. *Statistical Methods in Medical Research*, 27:1056 – 1066, 2018.

[8] M Kragh, J Rimestad, J Berntsen, and H Karstoft. Automatic grading of human blastocysts from time-lapse imaging. *Comput. Biol. Med.*, 115:103494, 2019.

[9] O Valls Murcia. A comprehensive probabilistic model for the embryo selection problem. Master's thesis, Technical University of Catalonia, 2021. URL `http://hdl.handle.net/2117/340945`.

[10] C Coughlan, W Ledger, Q Wang, F Liu, A Demirol, T Gurgan, R Cutting, K Ong, H Sallam, and T C Li. Recurrent implantation failure: definition and management. *Reproductive BioMedicine Online*, 28(1):14–38, 2015.

[11] 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)*, 39(1):1–38, 1977.

[12] J Herndez-Gonzlez, I Inza, and J A Lozano. Weak supervision and other non-standard classification problems: A taxonomy. *Pattern Recogn. Lett.*, 69:49–55, 2016.

[13] T Fawcett. Introduction to ROC analysis. *Pattern Recogn. Lett.*, 27:861–874, 2006.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210140

Towards Expert-Inspired Automatic Criterion to Cut a Dendrogram for Real-Industrial Applications

Shikha SUMAN [a,1], Ashutosh KARNA [a] and Karina GIBERT [a]

[a] *Knowledge Engineering and Machine Learning Group at Intelligent Data Science and Artificial Intelligence Research Centre, Universitat Politècnica de Catalunya*

Abstract.
Hierarchical clustering is one of the most preferred choices to understand the underlying structure of a dataset and defining typologies, with multiple applications in real life. Among the existing clustering algorithms, the hierarchical family is one of the most popular, as it permits to understand the inner structure of the dataset and find the number of clusters as an output, unlike popular methods, like k-means. One can adjust the granularity of final clustering to the goals of the analysis themselves. The number of clusters in a hierarchical method relies on the analysis of the resulting dendrogram itself. Experts have criteria to visually inspect the dendrogram and determine the number of clusters. Finding automatic criteria to imitate experts in this task is still an open problem. But, dependence on the expert to cut the tree represents a limitation in real applications like the fields industry 4.0 and additive manufacturing. This paper analyses several cluster validity indexes in the context of determining the suitable number of clusters in hierarchical clustering. A new Cluster Validity Index (CVI) is proposed such that it properly catches the implicit criteria used by experts when analyzing dendrograms. The proposal has been applied on a range of datasets and validated against experts ground-truth overcoming the results obtained by the State of the Art and also significantly reduces the computational cost.

Keywords. Hierarchical Clustering, Cluster Validity Indices, Calinski-Harabasz index, Dendrogram

1. Introduction

Hierarchical clustering is a powerful technique that very well addresses the challenge of discovering the underlying structure by creating a hierarchy of data partitions into smaller object groups from top to bottom. This is represented in a tree diagram, called *dendrogram*. The dendrogram provides a visual trace of the whole clustering process that the clustering experts can inspect manually and decide the number of clusters in the dataset. There are two clear advantages of this approach. Firstly, the expert takes an

[1] Corresponding Author: Shikha Suman, Knowledge Engineering and Machine Learning Group at Intelligent Data Science and Artificial Intelligence Research Centre, Universitat Politècnica de Catalunya, Catalonia, Spain; E-mail: shikha.suman@estudiantat.upc.edu

informed decision after inspecting the dendrogram closely. Secondly, it avoids running multiple runs of CVI's, such as *silhouette* [1], *gap-statistic* [2] as in the case of flat clustering methods which aims to find the best number of clusters corresponding to the local maxima.

The motivation behind this research comes from a wider research project of developing an *Intelligent Decision Support System* for *Industry 4.0* applications. The proposed system is expected to monitor the performance of 3D printers in real-time through sensor data. In [3], the authors describe a hierarchical clustering-based method to profile sensor data from 3D printing. A key challenge in this system is to find appropriate print profiles from the sensor data with no prior knowledge about the cluster formation or the number of clusters itself. Hence, the research discussed in this paper directly helps in building an automated solution to find the number of clusters similar to what the human experts would find using a dendrogram manually.

The previous work by Karna *et al.* [4], however, shows that the original *Calinski-Harabasz* index has several instances where the number of clusters disagrees with what human experts suggested. Although proposed $\Delta_{K_{cond}}$ criterion [4] improves performance but still requires further fine-tuning to correctly match with experts' criterion.

Hence, the goal of this paper is to assess the reasoning behind the disagreement between the human-criterion and the algorithmic method. To contribute to the research, a complete methodology is discussed with a new CVI for detecting the number of clusters automatically in hierarchical clustering. Its performance on 100 samples from a real-life dataset is also discussed in brief. The rest of the paper is thus structured as follows. A brief survey of related literature is presented in Section §2, followed by a formal definition of the research problem in Section §3. Two new CVI's are proposed in Section §4 and the corresponding methodology is discussed in detail in Section §5. A summary of the experimentation results is discussed in Section §6. The authors conclude the paper in Section §7 and discuss the future lines of research and use cases.

2. Literature Survey

Some strategies to determine the number of clusters in a hierarchical clustering are cross-validation, resampling, and finding the *knee* or *elbow* of an error curve. The cross-validation approach estimates the best number of clusters by partitioning the data into v parts and iteratively evaluate a cluster validity criterion developed on $v-1$ parts on the v^{th} part. However, the approach requires extensive computation which limits its applicability when the data becomes huge and results are expected quickly. In [5], Overall and Magee presented a replication-based stopping rule in which a replication defined by higher-order clustering helps identify the distinct underlying populations (clusters) in a multidimensional space. The resampling-based methods require drawing several bootstrapping samples from the parent distribution and this becomes infeasible as the size of the dataset grows and is not suitable when the size of the data is huge and time complexity is really important. Finding the number of clusters by optimizing a CVI curve and identifying the local maxima (or minima) at the *knee* is also used in a variety of situations. Sevilla *et al.* in [6] reviews several CVI's and how they associate with the data topology. *Gap-statistic* proposed by Tibshirani *et al.* [2] is another CVI. It tests the hypothesis that the model has a single cluster ($K = 1$) and tries to reject it with an alternative hypothesis that ($K > 1$). In [7], the authors used *clustering-gain* as a metric for finding an optimal number of clusters in hierarchical clustering. The *clustering gain curve* is designed to discover the distinct clusters when the intra-cluster similarity is the maximum and the inter-cluster similarity is minimum. In [8], Zhou *et al.* proposes a new CVI, called *CSP* (compact-separation-proportion) based on the idea of nearest neighbour. The optimal number of clusters is estimated corresponding to the maximum average value of the *CSP* index. The *Calinski-Harabasz* index [9] is one of the most common and often regarded as the best CVI to determine the number of clusters in hierarchical clustering. Milligan in [10] conducted an extensive experiment on thirty different CVI's and concluded the *Calinski-Harabsz* to be the most consistent one. In the recent work by Karna *et al.* in [4], the authors performed empirical analysis on several real-life datasets and presented an improved CVI, called, $\Delta_{K_{cond}}$, that maximizes the difference of successive *Calinski-Harabasz* indices over a range of K clusters ($k = 1, 2, ...K$) However, many instances were seen where this proposed criteria did not comply with the experts' determined number of clusters using dendrogram.

In [11] a proposal to find the right number of clusters in a big data environment is provided by Luna *et al.* that consists of two clustering validity indices handling a large amount of data in low computational time. The idea of reasoning over the heights of the nodes of the dendrogram has been explored by some authors, but none provides a simple and computationally cheap criterion that can suitably match what experts do in real practice.

To the extent of interpreting cluster patterns, several works relevant for this research are studied. In [12], the authors present an approach to interpret cluster patterns in real datasets. Gibert *et al.* [13, 14] the distance for clustering complex datasets with messy data is presented. In [15] this is generalized to include semantics variables. These metrics will be introduced when prior knowledge on the 3D printing problem enters into the system. In [16] the dynamics of the system is introduced to see how clusters evolve along time.

3. Research Problem

Let us consider a multivariate numerical dataset, with the information about a set I of N *k-dimensional* objects as $i_1, i_2, ...i_N$. Thus the goal of a hierarchical clustering is to partition I into a sequence of nested partitions P_k (k=2, 3, ...K= *N-1*).

$$P_k = \{C_{k_1}, C_{k_2}, ...C_{k_k}\}; k = 1, 2, ...N-1 \qquad (1)$$

where C_{k_k} represents the k^{th} cluster of the P_k partition of I.

The successive P_k are composed of disjoint clusters covering I. Thus, the dendrogram maps into the sequence $P_1, P_2, ..., P_{N-1}$ and $\forall k \in (2, 3, ...N-1), P_k$ is nested in P_{k-1} so that one of the clusters of the P_{k-1} subdivides in two in the P_k.

The objective of this paper is to develop an automatic criterion to identify the most appropriate P_k partition that divides the dataset I into k clusters such that the outcome is closest to what the human experts would achieve using the visual method. The main optimization criterion in this method is to find the value of k that optimizes the difference between the homogeneity of clusters and distinguishability among them. The more homogeneous and distinguishable the clusters are, the better is the partition.

4. Research Proposal

In theory, the number of clusters obtained by using *Calinski-Harabasz* method should match with the number of clusters deduced by the experts looking at the dendrogram. In [4], 5 different criteria based on *Calinski-Harabasz* index are proposed and evaluated to this purpose. In this research, all of them are evaluated over several datasets to see if they approach sufficiently well the criteria used by experts in visual inspection. In practice, a human expert usually decides the best cut of the dendrogram where the branches have longer gaps between nodes (regarding height), and each branch below the horizontal cut of the dendrogram results in a separate cluster. Underperformance has been detected on all these criteria and will be discussed in the application section. Experts choose the height representing the biggest disruption on the distinguishability. Following this intuition, two new criteria are presented in this paper to find the number of clusters based on the height of different nodes of a dendrogram.

Let $h_k, k \in 1 : N-1$ be the height of node k in a given dendrogram built over I. The values of h keep the value of the distance between clusters merged at each node of the dendrogram. These values directly depend on the linkage method used in the hierarchical process that generated the dendrogram.

I Δ_H **criterion**: The Δ_H criterion maximizes the gap of linkage height between two consecutive nodes of the dendrogram, starting from the root of the tree. Mathematically, the criterion can be defined as in eq 2.

$$K^*_{\Delta_H} = \underset{2\leq k\leq K}{\operatorname{argmax}}(\Delta_{H_k}); k \in (2,3,..K-1) \tag{2}$$

, where

$$\Delta_{H_k} = h_k - h_{k+1}; k \in (2,3,..K-1) \tag{3}$$

As it will be seen later in this paper, experimental results elicited that Δ_H criterion underperforms where the best cut of the tree is 2 clusters, as experts apply a heuristic in these cases. For this reason, a knowledge-based heuristic is introduced and a second criterion is proposed.

II $\Delta_{H_{cond}}$ **criterion**: The $\Delta_{H_{cond}}$ incorporates the heuristic that in some cases the experts skip a best cut in 2 clusters. This does not happen when the second-best cut is much closer to the bottom of the tree. This notion is represented through a ratio between the heights of the two highest nodes of the dendrogram, represented by h_{root} and h_2 respectively. Mathematically, this can be defined as eq 4.

$$K^*_{\Delta_{H_{cond}}} = \begin{cases} K^*_{2\Delta_H} & \text{if } K^*_{\Delta_H} = 2 \text{ and } (h_2/h_{root}) > 1/3 \\ K^*_{\Delta_H} & \text{otherwise} \end{cases} \tag{4}$$

where $K^*_{\Delta_H}$ and $K^*_{2\Delta_H}$ are the maximum and second maximum of Δ_K criterion. This introduces the flexibility to even consider 2 as the best clustering solution but only when the height of the root of the dendrogram (that results in two clusters) is at least thrice highest than the height of the second-highest node of the tree. Fig 1 and fig 2 help understand the distinction between the two scenarios. Details of these figures and the underlying logic will be clarified in section §6.

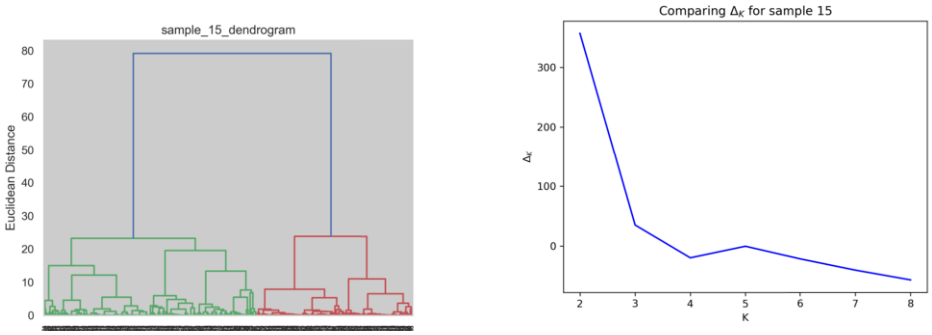


Figure 1. (a) Dendrogram for sample 15; (b) Δ_K curve for sample 15

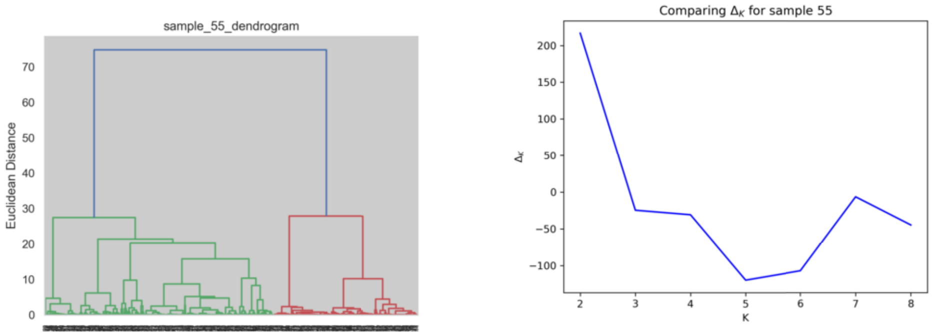


Figure 2. (a) Dendrogram for sample 55; (b) Δ_K curve for sample 55

The procedure to compute Δ_H and $\Delta_{H_{cond}}$ criteria are summarised as follows:

i For a dataset I containing N objects, compute the pairwise distance matrix.

ii Perform the hierarchical clustering I and build the corresponding dendrogram τ.

iii Obtain a list L of the nodes of τ sorted by descending height of nodes in descending order such that $H = h_{root} > h_2 > h_3 > .....h_{n-1}$.

iv Fix the maximum depth of nodes ($K \in 2 : N-1$) in the dendrogram to be considered to determine the number of clusters. Since, the bottom part of the dendrogram consists of nodes extremely close to each other, the optimal line of cut is placed in the top portion of the tree. Hence, in general, $K < N/2$ and this helps save half of the computations. The parameter K can heuristically be chosen (say, $K < N/2$). Return the first K nodes in L with their corresponding heights, namely (L', h'), in order to make a decision.

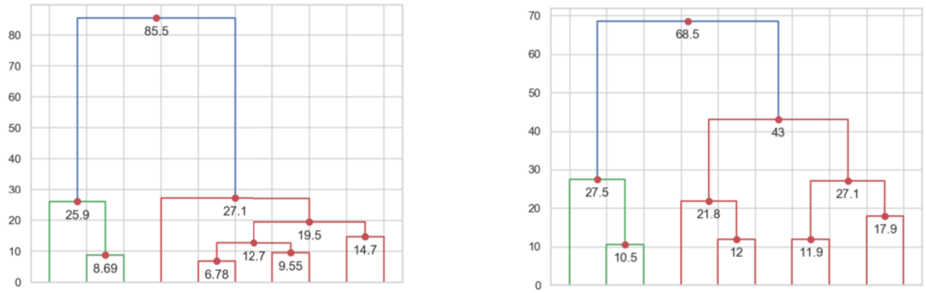


Figure 3. (a) Dendrogram of sample 10; (b) Dendrogram of sample 31

v Apply the criterion to the pair (L', h').

The approach can be verified by visualizing an annotated top part of τ with the height of top K nodes from the root of the tree. This step is to be carried out only to compare the values recommended by the criteria against the human-assigned one. An illustrative example can be seen in fig 3.

5. Research Methodology

This research evaluates 5 different CVI including the original *Calinski-Harabasz* index (denoted as M_K),its two variants (Δ_K, $\Delta_{K_{cond}}$) (as proposed in [4]) and the two new proposed criteria based on the inner morphology of the dendrogram, namely, Δ_H and $\Delta_{H_{cond}}$. These criteria are all applied to S datasets ($s \in (1, 2, ..S)$), all with a same number of objects N. For each dataset the dendrogram is obtained by using any hierarchical clustering method. The first K cuts of the dendrogram are obtained $K, k \in (2, 3, ...N/2)$. Given a CVI, f, $f \in (M_K, \Delta_K, \Delta_{K_{cond}}, \Delta_H, \Delta_{H_{cond}})$, a matrix $\chi_{f_s,k}$ is created where $\chi_{f_s,k}$ is the value of f for P_k of sample s.

Our main goal is to develop a criterion that approaches the number of clusters given by human experts. The proposed strategy is as follows:

I Let us consider a total of S dendrograms and the corresponding reference data, all of the same size.

II Subject each sample to hierarchical clustering (in this work with *Euclidean* distance and *Ward's* method), and obtain the dendrogram ($\tau_s, s \in (1, 2, ...S)$).

III For each τ_s, obtain P_k ($k \in 2...K = 9$) and compute the following for each k:

  i Compute the matrix χ for the *Calinski-Harabasz* index ($\chi_{f_s,k} = M_{s,k}$)

  ii Compute

$$\Delta_{s,k} = M_{s,k} - M_{s,k+1}, s \in 1, 2, ..S, k \in 2, 3, ..K - 1 \tag{5}$$

IV Let $K_{s,f}, f \in (M_k, \Delta_k, \Delta_{k_{cond}}, \Delta_H, \Delta_{H_{cond}}), s \in (1, 2, ..S), k \in (2, 3, ...9)$ be the returning number of clusters by each of the criteria f respectively.

V Get experts' assistance in providing the number of clusters from the dendrogram ($\tau_s, s \in (1, 2, ...S)$). Let $E_s, s \in (1, 2, ..S)$ be the number of clusters provided by the human experts for case $s \in (1, 2, ...S)$, by visual inspection of τ_s.

VI Compare the algorithmically determined number of clusters ($K_{s,f}$ against E_s) and assess the quality of f for each case $s_{\in}(1,2,...S)$

$$\varepsilon_{s,f} = |K_{s,f} - E_s| \tag{6}$$

VII Build a table of frequencies of $\varepsilon_{s,f}$ and the associated bar-charts and analyze the cases of largest mismatches.

VIII Take a representative case with $\varepsilon_{s,f} > 0$, visualize the dendrogram, get the CVI-proposed number of clusters and the one proposed by the experts, and try to understand the reason for the discrepancy by analyzing the dendrogram. Use as many cases as required until a comprehension of the failure of the criterion emerges. Use the results of this analysis to perform a modification on the CVI and evaluate the impact on the performance.

IX Compute the accuracy for a criteria f over S cases, denoted as A_f as follows:

$$A_f = \frac{card\{|K^*_{f,s} - E| = 0\}}{S}; s \in 1,2,..S \tag{7}$$

where $K^*_{f,s}$ denotes the optimal value of number of clusters using f^{th} criterion over s samples.

X Compare accuracy A_f over all of the candidate criteria. The winner criterion is the one that maximizes the accuracy.

6. Application

The research proposal has been evaluated on $S = 100$ real-life data samples obtained from an *Industry 4.0* process. Each dataset is of size N=500 and 10 numerical variables are used. *Euclidean* distance and *Ward* method have been used to cluster the samples. An ample mix of different types of morphological structures in their dendrograms is provided. In particular, it might be interesting to draw S random samples of a single reference dataset I, all of the same size, without replacement.

Following the methodology presented in 5, the dataset χ is built with 100 *Calinski-Harabasz* index curves for a range of $K = 8$ clusters (each curve representing a dendrogram with a different topology as shown in fig 4.

It can be seen that different morphologies of dendrograms correspond to different patterns of curves. Fig. 1 and fig. 2 show how the pattern of the Δ_K curve seems to be associated with the good or bad performance of the criterion, and also with different morphology of the reference dendrogram itself. Fig. 1 shows Δ_K curve monotonically decreasing with the maxima occurring at 2 ($K^*_\Delta = 2$), and the same is also evident from the dendrogram as well ($E_{15} = 2$), resulting in accurate match ($\varepsilon_{15,\Delta_K} = 0$). On the contrary, in the case of fig.2, while the local maximum occurs at 2 ($K^*_\Delta = 2$), the experts skip 2 as the best solution and rather suggests 7 clusters ($E_{55} = 7$) and results in a big mismatch ($\varepsilon_{55,\Delta_K} = 5$) which in fact, is the second local maximum.

Hierarchical clustering was performed on the dataset of Δ_K curves of all 100 samples in order to find groups of similar dendrogram morphologies together for detailed analysis. Seven distinct classes are identified with the clustering exercise and samples falling in different classes are analyzed independently. It is observed that the practice of skipping the maxima when (K=2) and switching to the second optimal value of k is quite common among human experts. This is due to an implicit exercise that cutting the tree into 2 clusters leads to a dichotomous solution which is rarely useful enough for further decision making. And thus, an implicit clustering rule often tends to shift to second-best solution where more than 2 clusters exist. This reasoning can also be seen when the authors in [4], proposed $\Delta_{K_{cond}}$ criterion. In a particular case of sample-55 (fig 2), $\Delta_{K_{cond}}$ does provide 7 as the number of clusters matching correctly with the experts. However, this is not always true and while analyzing the classes from the previous clustering exercise, several instances can be seen where this condition disagrees with the experts. Hence, it can be concluded that the *Calinski-Harabasz* criteria either in its original form or the variants as proposed in [4], do not truly capture the structure of the dataset, whereas the experts make a decision based on the vertical gaps in the dendrogram. This leads us to our proposal of using the height of nodes in a dendrogram to decide the best number of clusters.

The error distribution after applying the Δ_H criterion on all 100 samples can be seen in Table 1. It is clear that in most of the cases, $K^*_{\Delta_H} = 2$, that implies that in 85% of the cases, the algorithm determined 2 as the best cluster while the expert determined otherwise. This is similar to the case of *Calinski-Harabasz* index (M_K) which differs from the experts as guessed. This is aligned with the experts' judgment as discussed in the case of sample-15 and sample-55.

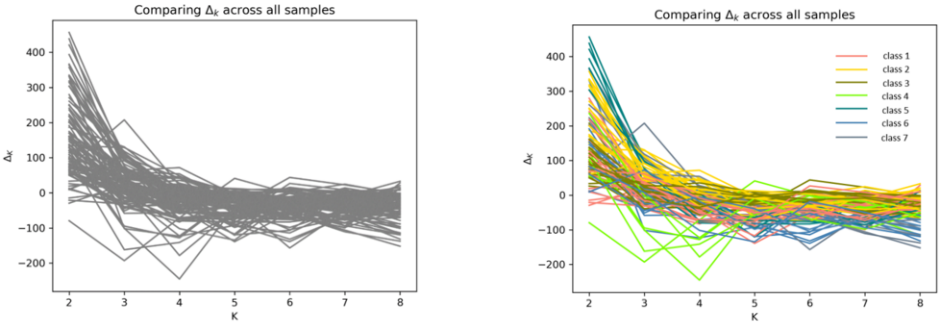


Figure 4. (a) CH curves for all samples; (b) CH curves for all samples after clustering

Following the proposal in section 4, the $\Delta_{H_{cond}}$ can visually be expressed with the help of an annotated dendrogram of sample-10 and sample-31. From fig 3, one can compute the height-factor as ($h_2/h_{root} = 27.1/85.5 = 0.316$) and also $K^*_{\Delta_H} = 2$. Hence, following the criterion in eq 4, the $K^*_{\Delta_{H_{cond}}} = 2$ as the ratio of height is lesser than $1/3$.

Applying the same criterion to sample-31, the height-factor becomes ($h_2/h_{root} = 43.0/68.5 = 0.627$) and with $K^*_{\Delta_H} = 2$ and $K^*_{2\Delta_H} = 3$. Therefore, the second maxima is to considered and thus $K^*_{\Delta_{H_{cond}}} = 3$ is chosen as the best value. Thus, the proposed criterion $\Delta_{H_{cond}}$ correctly matches with experts' number under different morphological structure of the dendrogram.

The error distribution of the $\Delta_{H_{cond}}$ criterion along with other candidate criteria, is provided in Table 1. This method correctly matches 93% of all experts' numbers and performs significantly greater than all other criteria including the proposals in [4].

Table 1. Error distribution of the different CVI analyzed

	Values of error									
CVI	$\varepsilon = 0$	$\varepsilon = 1$	$\varepsilon = 2$	$\varepsilon = 3$	$\varepsilon = 4$	$\varepsilon = 5$	$\varepsilon = 6$	$\varepsilon = 7$	**Error rate**	**Acc.**
M_K	15	23	9	5	15	13	19	1	0.85	0.15
Δ_K	23	38	16	15	5	2	1	0	0.77	0.23
$\Delta_{K_{cond}}$	55	25	10	6	2	2	0	0	0.45	0.55
Δ_H	15	44	17	15	5	3	1	0	0.85	0.15
$\Delta_{H_{cond}}$	93	2	2	3	0	0	0	0	0.07	0.93

7. Conclusion

The research reveals that though in theory, the *Calinski-Harabasz* index, being the ratio of between-cluster and within-cluster variance, works similar to how a dendrogram is built when choosing $Ward's$ criteria, it does not fully unveil the properties of the data. As a result, the number of clusters determined using dendrogram varies to a large extent when the same is being done through the M_K method. A deeper analysis of this has been discussed in section §6, which leads to developing a novel approach of using dendrogram-height based index ($\Delta_{H_{cond}}$) that emerges to be far superior to any of the criteria listed here. A key advantage of using $\Delta_{H_{cond}}$ lies in the direct application of scaling the hierarchical clustering into real-life applications where humans are replaced with intelligent automated systems while still retaining their inherent heuristic in a mathematical form (as defined in eq 4).

This research directly fits into a bigger project that aims at developing an intelligent decision support system for Industry 4.0 processes in which a specific module deals with the automatic clustering part. Considering the immediate application of this approach, certain preprocessing steps have been ignored in this research such as missing-value-treatment and outlier-removal, however, they may be included in the future lines of this research. Also, the research methodology has been applied on real-life datasets with unknown clusters and rather experts' judgment and is taken as ground truth. This would be extended further in the future by applying the proposal on synthetic as well as a pre-labeled dataset to assess the performance of this criterion. It is also to be noted that the main intent of this research is to identify the number of clusters closest to a human, however, a deeper investigation is also to be done in the future with respect to assessing the quality of the clusters. A key advantage of this research lies in reducing the CPU-time that is consumed in finding the *elbow* of the CVI curve. The proposed criterion identifies the right number of clusters from the initial linkage matrix and no multiple runs of hierarchical clustering are needed.

References

[1] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[2] Robert Tibshirani, Guenther Walther, and Trevor Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423, 2001.

[3] Ashutosh Karna and Karina Gibert. Using hierarchical clustering to understand behavior of 3d printer sensors. In *International Workshop on Self-Organizing Maps*, pages 150–159. Springer, 2019.

[4] Ashutosh Karna and Karina Gibert. Automatic identification of the number of clusters in hierarchical clustering. *Neural Computing and Applications*, pages 1–16, 2021.

[5] John E Overall and Kevin N Magee. Replication as a rule for determining the number of clusters in hierarchial cluster analysis. *Applied Psychological Measurement*, 16(2):119–128, 1992.

[6] Beatriz Sevilla-Villanueva, Karina Gibert, and Miquel Sànchez-Marrè. Using cvi for understanding class topology in unsupervised scenarios. In *Conference of the Spanish Association for Artificial Intelligence*, pages 135–149. Springer, 2016.

[7] Yunjae Jung, Haesun Park, Ding-Zhu Du, and Barry L Drake. A decision criterion for the optimal number of clusters in hierarchical clustering. *Journal of Global Optimization*, 25(1): 91–111, 2003.

[8] Shibing Zhou, Zhenyuan Xu, and Fei Liu. Method for determining the optimal number of clusters based on agglomerative hierarchical clustering. *IEEE transactions on neural networks and learning systems*, 28(12):3007–3017, 2016.

[9] Tadeusz Caliński and Jerzy Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974.

[10] Glenn W Milligan. A monte carlo study of thirty internal criterion measures for cluster analysis. *Psychometrika*, 46(2):187–199, 1981.

[11] José María Luna-Romera, Jorge García-Gutiérrez, María Martínez-Ballesteros, and José C Riquelme Santos. An approach to validity indices for clustering techniques in big data. *Progress in Artificial Intelligence*, 7(2):81–94, 2018.

[12] Beatriz Sevilla-Villanueva, Karina Gibert, and Miquel Sànchez-Marrè. A methodology to discover and understand complex patterns: Interpreted integrative multiview clustering (i2mc). *Pattern Recognition Letters*, 93:85–94, 2017.

[13] Karina Gibert and Claudio Ulises Cortés García. Weighting quantitative and qualitative variables in clustering methods. *Mathware & soft computing. 1997 Vol. 4 Núm. 3*, 1997.

[14] Karina Gibert and Ramon Nonell. Impact of mixed metrics on clustering. In *Iberoamerican Congress on Pattern Recognition*, pages 464–471. Springer, 2003.

[15] Karina Gibert, Aïda Valls, and Montserrat Batet. Introducing semantic variables in mixed distance measures: Impact on hierarchical clustering. *Knowledge and information systems*, 40(3):559–593, 2014.

[16] Karina Gibert, Gustavo Rodríguez-Silva, and Ignasi Rodríguez-Roda. Knowledge discovery with clustering based on rules by states: A water treatment application. *Environmental Modelling & Software*, 25(6):712–723, 2010.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210141

Limits of Conventional Machine Learning Methods to Predict Pregnancy and Multiple Pregnancy After Embryo Transfer

Núria CORREA[a,1], Rita VASSENA[a], Jesús CERQUIDES[b], Josep Lluís ARCOS[b]
[a] *Clinica Eugin-Eugin Group*
[b] *IIIA-CSIC*

Abstract. When training models to learn the relationship between two or more variables, we expect to see previously demonstrated knowledge about that relationship reflected in the resulting estimators. For some domains, such as healthcare, it is imperative for actual implementation of those models that their predictions respect this knowledge. In this study we focus on Assisted Reproduction Technology (ART), the subspecialty of gynecology occupied with treating human infertility, and where the goal of any treatment is the delivery of a healthy newborn. A common ART treatment is In vitro Fertilization (IVF), where embryos are generated in vitro from collected sperm and oocytes, and transferred to the uterus of the patient after selecting those most likely to give rise to a healthy pregnancy. IVF has an approximate 30% successes rate per cycle; to palliate for this low success rate, a common practice so far has been to transfer two embryos simultaneously, aiming to increase the chances of a favorable outcome. While increasing overall live birth rates, this method has also led to an alarmingly high rate of twin and triplet births, associated with four times higher risk of perinatal mortality and increased obstetric complications. Our objective is to predict the chances of both pregnancy (P) and multiple pregnancy (MP) following either single embryo transfer (SET) or double embryo transfer (DET), and in so facilitating an informed decision on how many embryos to transfer. From existing literature, it is known that: (1) it is not possible for the chances of both P and MP to be decreased by increasing the number of embryos; (2) MP chances cannot be higher than P; and (3) chances of pregnancy are highly correlated with age, embryo stage, and quality. With a dataset generated from an existing observational study, we trained several state-of-the-art classifiers to predict P and MP given SET and DET. Analyzing the results, all classifiers achieved promising AUC scores. However, Random Forest and Gradient Boosting predicted negative chance differences in many instances when increasing the number of embryos infringing the first constraint. Logistic Regression predicted always positive differences, but in some instances it infringes the second constraint, predicting higher chances of MP than of P. Moreover, it showed little to no variation across ages or embryo stages violating third constraint. Conventional Machine Learning models struggle to reflect the real-world outcomes when using DET versus SET in specific patients. More informative variables could help, but it is already worrisome that variables as important as age and embryo stage do not result already in any variation, and that when models do show variation, in many cases they predicted decreasing chances of success with more embryos. We conclude that new and different approaches are needed to correctly model this scenario and, likely, many others resembling this one.

Keywords. Machine Learning, Classification, Healthcare

[1] Núria Correa, R+D Department, Clínica Eugin, Balmes 236, 08006 Barcelona, Spain; E-mail: ncorrea@eugin.es.

1. Introduction

With the rising general popularity of Artificial Intelligence (AI) and, specifically, Machine Learning (ML), several fields have jumped on the bandwagon of applying them to different processes. One such field is healthcare, where many high stakes and fast decisions must be made. As high dimensional data registers are frequently available, it stands to reason that those could be learned from using ML. In many healthcare scenarios a heavy research background already exists, providing a high amount of evidence-based knowledge. In this context, it is expected that previously demonstrated data relations are picked up by trained models and their predictions heed them. Additionally, to ensure user confidence, the explainability of ML models is of paramount importance. Explainability also needs to be coherent with previously demonstrated knowledge. In other words, expectations on how the models will work are set by preceding research, and failure to comply with them can diminish the confidence of the users in the models' predictions.

This is the situation of our subject of interest: Assisted Reproductive Technologies (ART), a subspecialty of gynecology that is preoccupied with the instrumental treatment of human infertility and whose main goal is the delivery of a healthy newborn. In order to achieve this objective, different techniques have been developed and are applied depending on the necessities of the patient. A common kind of ART treatment is In Vitro Fertilization (IVF), where oocytes and sperm are combined in vitro to generate embryos. After selecting those expected to have better chances of giving rise to a healthy pregnancy, the embryos are transferred to the uterus of the patient. IVF provides an approximated 30% pregnancy rate per treatment, which leads to about 20% delivery rate [1] . These rates can undoubtedly be frustrating for both professionals and patients. To mitigate low success rates, the transfer of two embryos simultaneously to the uterus has been proposed. This certainly increases the chances of achieving a pregnancy versus Single Embryo Transfer (SET) [2] ; Double Embryo Transfer (DET) now represents 54.5% of all embryo transfers. Unfortunately, the increase in success comes with an increased obstetrical risk, reflected by the troublingly high 17% of twin births DET. Measured against singleton births, twin births have a four times higher risk of perinatal mortality. Twin pregnancies are also associated with an increased risk of obstetric complications, higher rates of miscarriage, pregnancy-induced hypertension, gestational diabetes, premature labor and abnormal delivery compared to singleton pregnancies [3] . As a consequence, a twin pregnancy is an undesired outcome of ART cycles.

Nevertheless, the rate of DET remains high; why is this? The issue is indeed complex. As stated before, Randomized Controlled Trials (RCTs) have consistently shown that SET provides lower pregnancy rates than DET, but they do so with the bonus of a much lower twin rate. Literature also indicates that the cumulative pregnancy rate between repeated SET and a single round of DET is similar, but there is a much lower twin rate in patients that get SET+SET vs. DET [2]. This would, from a strictly clinical point view, lead to an easy solution, which would be to always use repeated SET. But, as stated before, the issue is not that straightforward.

On the one hand, we should acknowledge that the embryos available to a woman for transfer are not always of high morphological quality, and having worse morphology is an indicator of worse development potential and higher aneuploidy rates [4]. In these cases, DET is used as a strategy to allow for higher pregnancy rates in bad prognosis treatments, assuming that the risk of multiple pregnancy should be lower as one of the two embryos transferred has low chances to implant. Further, embryo stage may influence the outcome, as there is moderate quality evidence that blastocyst stage embryos (at day 5 or 6 after fertilization) have better chances of pregnancy versus cleavage stage embryos (at day 2 or 3 after fertilization) [5]. Also, regardless of embryo quality and stage, the specifics of every case modulate the chances of pregnancy as does for example the age of the oocyte [6] and its origin (donor or own oocytes), the integrity of the uterine environment and shape, the reproductive history of the couple or single patient, the parameters and origin (donor or partner) of the semen used to fertilize the oocytes, etc. On a day-to-day basis, all this information is processed by the clinical experts in order to make a professional recommendation based on literature and hands-on experience on the adequate number of embryos to be transferred in order to achieve the highest possible live birth rates with the lowest possible multiple pregnancy rate.

On the other hand, patients are paramount in these processes, as they are the ones going through the treatment with the very emotionally charged goal of being able to give birth. They participate actively in making the final decision of how many embryos will get transferred, and often non-clinical factors weight in their decision. Some of those factors include their psychological state (affected by repeated treatments, urgency to get pregnant, previous interrupted pregnancies, etc.), the economic pressure of the treatments and the information that they receive and/or understand [7].

Considering all this, it is clear that the clinical objective when selecting between a SET or DET treatment for each individual patient is to get the highest pregnancy chance with the lowest twin pregnancy risk. And so, it is natural to search for methods that allow us to predict better the chance of pregnancy (P) and multiple pregnancy (MP) for patients before getting SET or DET. Here is where ML can be of help.

Then, the technical objective of this study is to train models able to predict chances of P and MP given a set of covariates that include both treatment options. Getting accurate models for these tasks would enhance professionals' confidence in aiding patients to make an informed decision. But in order for those models to be really regarded as usable in clinical practice they need to heed previously demonstrated knowledge, leading us to identify three main constraints:

1. Under stable conditions (same patient, same cohort of embryos) it is not possible for the chances of both P and MP to be decreased by increasing the number of embryos transferred.

2. Under stable conditions MP chances cannot be higher than P.

3. Chances of P and MP are highly correlated with age, embryo stage, and quality.

To properly test the performance of conventional ML models not only standard measures as AUC need to be analyzed, but also compliance with all three constraints needs to be examined.

Few studies have been carried out in this regard, but the ones that did give us some interesting insight. A very thorough report on the theme performed [8] recounts construction of P and MP models using first UK national reports with AUC 0.60 for the first model and 0.66 for the second, and then information from multiple private centers with more predictor variables and slightly better AUC scores. It also reviews an approach modeling separately the uterus component and the embryo component. Another study creates only an MP model for patients that got DET [9]. Lastly, another interesting study created independent models: one for P and MP on DET cycles, and another one for P on SET cycles, getting AUCs between 0.64 and 0.75 [10]. Interestingly, this last model has been also tested on patients and has helped to significantly reduce the incidence of MP. All of these studies are promising, but do not check for the first two constraints which we find are certainly critical.

2. Materials and methods

There are multiple public and published sources that report results on pregnancy and multiple pregnancy with both SET and DET [8 , 11]. All these populational studies are coherent between them but offer only summarized sample statistics, and no granular patient level datasets are publicly available. In this study, to ensure reproducibility, we focused on the data from the observational study by Aldemir et al. (2020) [11], taken as a guiding example to synthetically generate a dataset. In their study where 2298 patients were included three groups were compared: those who got DET with good quality embryos (GQEs), DET with mixed quality embryos (MQEs), and SET with good quality embryos. For those three groups several variables were gathered, including age, embryo stage, pregnancy and multiple pregnancy.

The replicated dataset was carefully constructed. Maternal age was simulated for every group using mean and standard deviation reported by the observational study to randomly sample from a normal distribution, resulting in 33.28 ± 4.1 for the first group, 34.4 ± 3.8 for the second and 29.2 ± 4.1 for the third. Individual outcomes of P and MP per group and embryo stage were sampled randomly from reported results using a Bernoulli distribution. The resulting proportions, shown in Table 1 and 2, had less than a 5% deviation compared to the original study results. Further, strict restrictions were put in place in order to avoid inconsistencies on our artificial dataset, such as cases with positive MP results but a negative P result.

Table 1. Proportions of pregnancy and multiple pregnancy instances by group in patients who got transferred embryos at the cleavage stage

	DET with GQEs **n=324**	**DET with MQEs** **n=127**	**SET with GQE** **n=887**
P	41.05	35.43	29.99
MP	23.46	9.45	3.16

Table 2. Proportions of pregnancy and multiple pregnancy instances by group in patients who got transferred embryos at the blastocyst stage

	DET with GQEs **n=174**	**DET with MQEs** **n=52**	**SET with GQE** **n=734**
P	56.32	23.08	43.46
MP	32.76	25.00	2.32

Three common ML classifiers were selected to be trained on our resulting database. Those classifiers were Logistic Regression (LR), Random Forest Classifier (RFC) and Gradient Boosting Classifier (GBC). 80% of the synthetic database was used to train them and the other 20% was reserved for testing purposes. Average AUC and accuracy scores were obtained by cross validating 10 times over the training dataset.

As not only conventional scores are important in this kind of scenarios, the predicted outcomes on the test portion were analyzed to assess compliance of the 3 stated constraints. In order to do that all patients (1) got predicted probabilities of P and MP with SET and DET separately to detect any negative "effects"; (2) got predicted probabilities of P and MP to detect cases with higher MP chances than those of P; and (3) both P and MP predicted chances were examined for its relations with maternal age and embryo stage and quality.

3. Results

After analyzing common scores as AUC and Accuracy, LR and GBC seem to be the ones that fare better at predicting both outcomes, with LR being slightly better at AUC and GBC at accuracy (see Table 3). Regarding the mean expected effect of using DET versus SET for every specific patient, all estimators get close to values described in literature regarding P, which fall between 12% and 23% increased chances [2]. This is not the case of MP, where multiple RCTs pooled suggest an increase between 11% and 13%. RFC and GBC are slightly over those values, and LR is very clearly out of the described range.

Table 3. Results of the divided by type of model (Logistic Regression, Random Forest Classifier, Gradient Boosting Classifier) and outcome (Pregnancy and Multiple Pregnancy).

	AUC	Accuracy	Mean effect	Constraint 1	Constraint 2	Constraint 3
LR-P	0.58	0.55	0.14	Yes	No	Partial
LR-MP	0.78	0.75	0.55	Yes	-	No
RFC-P	0.52	0.54	0.17	No	No	No
RFC-MP	0.71	0.86	0.24	No	-	No
GBC-P	0.56	0.62	0.12	No	No	Partial
GBC-MP	0.77	0.91	0.21	No	-	Partial

When considering the first constraint (under the same conditions increasing the number of embryos cannot decrease the success chances), only LR complies fully with it. RFC and GBC both show multiple instances where their predictions estimate a decrease in chances in DET vs SET in the same patient, as shown for example in Figure 1.

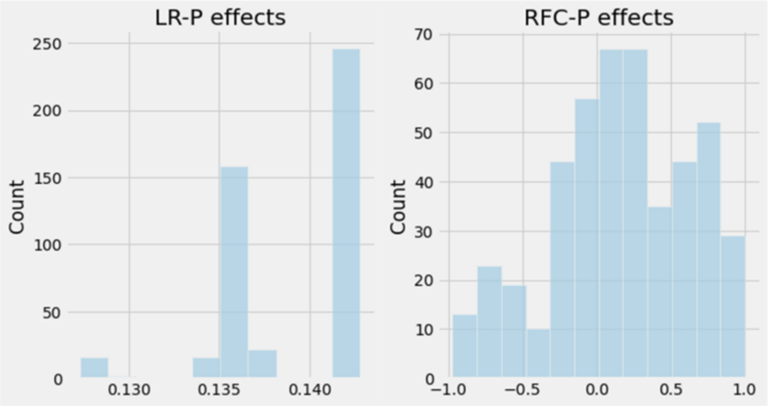


Figure 1. Probability differences between predictions on the same patients with SET and DET in the models Logistic Regression (left) and Random Forest Classifier (right) trained to predict pregnancy outcomes.

Looking upon the second constraint we found no compliance across all models studied, with GBC being the one with the least instances where the constraint was infringed (see Figure 2).

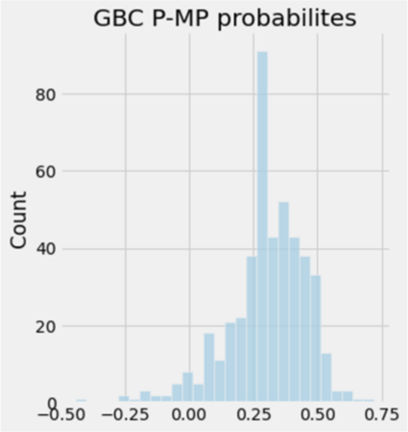


Figure 2. Distribution of the differences in predicted probabilities for pregnancy and multiple pregnancy using Gradient Boosting Classifiers.

Lastly, for the third constraint both LR and GBC comply only partially and RFC does not comply with it. It is referred as partial as with both models age seems to add little to no variation in predicted P when embryo stage does, as shown in Figure 3. Predicting MP though, GBC seems to show some variation across ages, but LR does not comply at all.

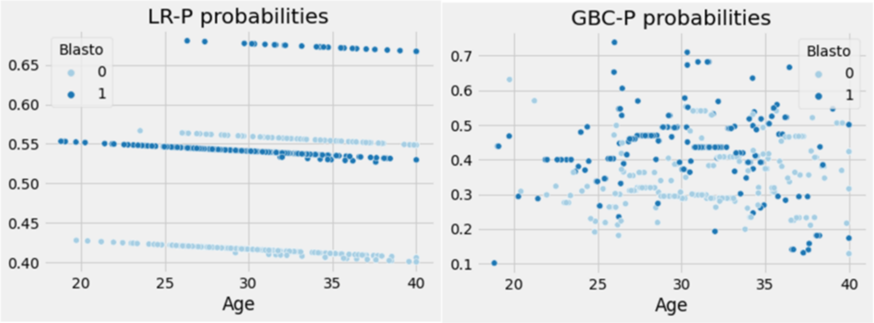


Figure 3. Logistic Regression and GBC pregnancy predicted probabilities plotted against maternal age and colored by embryo stage (blastocyst yes or no).

4. Discussion

We show here that the three conventional ML models tested are not entirely suitable for the task at hand, even if AUC scores remain close to those obtained in literature. The highest performing algorithm seems to be LR but even so, it only complies with one and a half of our presented constraints. If presented to a field expert for clinical practice, evidently it would be regarded as unfaithful and thus unusable. In any ML implementation related to healthcare, not only a model needs to be accurate, but it also needs to convince the professional about its reliability, and that means being consistent with evidence-based knowledge. Especially nowadays when AI and ML models are under public scrutiny and asked to be accountable, and that leads to be able to explain their decisions.

Concerning the first constraint, there seems to be an inbuilt bias in the dataset, where younger patients and embryos with better qualities or more advanced embryo stages tend to lead to more SET treatments. This is in agreement with previous knowledge of the field, as better prognosis is associated with a higher risk of MP, and so professionals and patients tend to prefer SET. Older patients and lower quality embryos tend to fair worse, and so with lower risks of MP, they tend to get more DET attempts. In other words, treatment is not randomized, as it is often the case in observational databases. Also, our dataset does not contain SET with embryos of worse quality, nor DET with both embryos of bad quality. This may create a confounding effect that cannot be accounted for correctly by the model. It would be interesting to identify from the literature studies with more types of embryo combinations, to understand if this may remain a concern. Unfortunately, none of the published researches check for that constraint.

As for the second constraint, one of the main problems in this approach to the matter seems to be the need to model two separate but closely related outcomes without being able to state some restrictions on how the models should predict both outcomes for the same patient. Even if treated as a multiclass problem (with outcomes failure, P, and MP) we would not be able to specify that there should never be a higher chance of MP than of P with common ML models. Looking at the available literature, a way of overriding the second constraint would be by constructing the MP model only using data of DET cycles that got a successful P, as that is what all studies do in constructing MP models. But that would drastically reduce the size of the available dataset and maybe hinder the models' performance. It also completely ignores the prediction of the probabilities of MP for SET cycles that, though they have very little chances in general of an instance of MP, could be also interesting to be able to predict.

Last but not least, the third constraint seems to be fairly complied with in previous studies on the matter where datasets include far more information, which would lead us to think that possessing a database of that characteristics would enable us to get models compliant with it.

5. Conclusions

In this work we have shown that conventional ML models, even when performing well in terms of prediction score at the population level, struggle considerably at the individual level. In doing so, they fail to comply with evidence-based derived constraints. As we stated in our motivation, in healthcare explainability is mandatory and it should always guarantee alignment with previous evidence-based knowledge. As exposed in other studies [12], failing to ensure cohesiveness can lead to diminished user confidence in the model and, in the worst-case scenario, to detrimental consequences for patients.

Focusing on our specific experiment, for the second and third constraints there seems to be possible solutions, but for the first one there seems not to be a straightforward answer. Finding a way to define beforehand the relationship between key variable treatment and outcome as monotonically ascending could take us a step closer to obtaining more realistic models. This challenge is not specific of the situation described here, rather it is somewhat endemic in the healthcare field, and so it constitutes a barrier to adopt AI solutions. Therefore, it is clear that new and different approaches to this kind of challenges would be needed.

Acknowledgments

This work was supported by Doctorat Industrial funded by Generalitat de Catalunya (DI-2019-24),by project CI-SUSTAIN funded by the Spanish Ministry of Science and Innovation (PID2019-104156GB-I00), and by intramural funding by Clinica Eugin-Eugin Group. Núria Correa is a Ph.D. Student of the doctoral program in Computer Science at the Universitat Autònoma de Barcelona.

References

[1] De Geyter C, Calhaz-Jorge C, Kupka MS, Wyns C, Mocanu E, Motrenko T, et al. ART in Europe, 2014: Results generated from European registries by ESHRE. Hum Reprod. 2018;33(9):1586–601.

[2] Kamath MS, Mascarenhas M, Kirubakaran R, Bhattacharya S. Number of embryos for transfer following in vitro fertilisation or intra-cytoplasmic sperm injection. Cochrane Database Syst Rev. 2020;2020(8).

[3] Crosignani PG, Baird DT, Barri P, Bryan E, Collins J, Diedrich K, et al. Multiple gestation pregnancy. Hum Reprod. 2000;15(8):1856–64.

[4] Hardarson T, Caisander G, Sjögren A, Hanson C, Hamberger L, Lundin K. A morphological and chromosomal study of blastocysts developing from morphologically suboptimal human pre-embryos compared with control blastocysts. Hum Reprod. 2003;18(2):399–407.

[5] Glujovsky D, Farquhar C, Quinteiro Retamar AM, Alvarez Sedo CR, Blake D. Cleavage stage versus blastocyst stage embryo transfer in assisted reproductive technology. Cochrane Database Syst Rev. 2016;2016(6).

[6] Grøndahl ML, Christiansen SL, Kesmodel US, Agerholm IE, Lemmen JG, Lundstrøm P, et al. Effect of women's age on embryo morphology, cleavage rate and competence - A multicenter cohort study. PLoS One. 2017;12(4):1–12.

[7] de Lacey S, Davies M, Homan G, Briggs N, Norman RJ. Factors and perceptions that influence women's decisions to have a single embryo transferred. Reprod Biomed Online. 2007;15(5):526–31.

[8] Roberts SA, McGowan L, Hirst WM, Brison DR, Vail A, Lieberman BA. Towards single embryo transfer? modelling clinical outcomes of potential treatment choices using multiple data sources: Predictive models and patient perspectives. Health Technol Assess (Rockv). 2010;14(38):1–237.

[9] Lannon BM, Choi B, Hacker MR, Dodge LE, Malizia BA, Barrett CB, et al. Predicting personalized multiple birth risks after in vitro fertilization-double embryo transfer. Fertil Steril [Internet]. 2012;98(1):69–76. Available from: http://dx.doi.org/10.1016/j.fertnstert.2012.04.011

[10] Vaegter KK, Berglund L, Tilly J, Hadziosmanovic N, Brodin T, Holte J. Construction and validation of a prediction model to minimize twin rates at preserved high live birth rates after IVF. Reprod Biomed Online [Internet]. 2019;38(1):22–9. Available from: https://doi.org/10.1016/j.rbmo.2018.09.020

[11] Aldemir O, Ozelci R, Baser E, Kaplanoglu I, Dilbaz S, Dilbaz B, et al. Impact of Transferring a Poor Quality Embryo along with a Good Quality Embryo on Pregnancy Outcomes in IVF/ICSI Cycles: a Retrospective Study. Geburtshilfe Frauenheilkd. 2020;80(8):844–50.

[12] Obermeyer Z, Powers B, Vogeli C, Mullainathan S. Dissecting racial bias in an algorithm used to manage the health of populations. Science (80-). 2019;366(6464):447–53.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210142

Collection, Processing and Analysis of Heterogeneous Data Coming from Spanish Hospitals in the Context of COVID-19

Marta BARROSO [a,1], Adrián TORMOS [a] , Raquel PÉREZ-ARNAL [a] ,
Sergio ALVAREZ-NAPAGAO [a] and Dario GARCIA-GASULLA [a]
[a] *Barcelona Supercomputing Center (BSC), Spain*

Abstract. The COVID-19 pandemic has already caused more than 150,000,000 cases worldwide. In Spain this has lead to a massive and simultaneous saturation of all sanitary regions. Coherently, the quick and consistent understanding of the COVID-19 disease requires of the combined analysis of thousands of medical records generated by dozens of different institutions. In the context of the publicly funded CIBERES-UCI-COVID project, we have gathered, cleaned and pre-processed data from heterogeneous sources – more than 30 hospitals, with different data entry systems – in order to produce a unified database, of more than 6.000 patients, that is used in several clinical studies being carried by different multi-disciplinary groups. In this paper, we identify the complexities we encountered, the solutions we applied, and we summarise the statistical and machine learning techniques we have applied for the studies.

Keywords. COVID-19, Machine learning, Data migration, Continuous development and integration (CD/CI), Automated report generation

1. Introduction

The COVID-19 pandemic has been the first humankind has faced as a completely digitised society. Thanks to that, developed countries had the resources to collect information related to COVID-19 efficiently and in real-time from early stages of the pandemic, resulting in the creation of multiple medical databases overnight. Clearly, the variety of information sources complicates the homogenisation and aggregation of data, two necessary steps for any automated analysis, mining or learning to be made on that data.

In May 2020, the CIBERES-UCI-COVID [9] project was awarded, funded by ISCIII. This project has the goal of carrying out an in-depth, retrospective and multicenter analysis on the distribution, correlations and missing values of covid-infected patient data in Spain. Within this project, artificial intelligence (AI) is used for extracting information about the factors involved in mortality, for classifying patients according to certain patterns, and for estimating the time for a group of individuals to experience an event of interest (*e.g.*, reach a critical condition or require mechanical ventilation), among other

[1] Corresponding Author: Marta Barroso, c/Jordi Girona, 31, 08034 Barcelona, Spain; E-mail: marta.barroso@bsc.es

things. To feed all this processes, data is gathered from many different hospitals in Spain, including several specific sources such as Getafe hospitals and the SEMICYUC consortium (which have their own data storage system). For gathering, processing and exploiting such amount of information, the collaboration of experts from interdisciplinary fields is required. For this reason the CIBERES-UCI-COVID consortium is composed by medical doctors, bioinformatics and AI researchers. The authors of this paper have the latter role, and were in charge of most of the automation process, as detailed next.

In this paper we review some of the methodologies used within CIBERES-UCI-COVID for two different purposes. First to collect, aggregate and summarise the available information in an accessible manner. And second to exploit this information through analysis, mining and learning methods for producing novel insights of interest. In detail, the contributions of the paper are the following:

- Describing the process to develop a complete and unified database derived from REDCap platform (a data gathering, form-based system). We also detail the structure and codification of original data and discuss the complexities of current representation in §2.
- For the sake or re-usability and accessibility, we review the technical requirements of the project, identifying key aspects and functionalities that our work must provide to the medical experts and the rest of involved institutions. In addition, for structuring information and data flows, several proposals have been taken into account. The selection criteria and the current proposal are discussed in §3.
- Developing a fast and efficient method to combine different sources into one database. We also detail how this database is populated and kept updated regularly by means of migrations. During this process we homogenise data in order to avoid inconsistency issues. This is explained in depth in §4.
- Automating data pre-processing capable of transforming any dataset within the context of CIBERES-UCI-COVID and generating reports on missing data, outliers, correlations and feature selection analysis. The aspects reported for each pre-processing task are explained in detailed in §5.
- Defining a global configuration for automate pre-processing, report generation process and to generate validated data with the appropriate format for subsequent studies. This is discussed in §6.

2. Data sources

REDCap [5] is a web-based database for medical and biomedical research support created by the REDCap Consortium [4]. Given the familiarity of the medical partners with this technology, the CIBERES-UCI-COVID project uses this database as single source of truth. This database is filled by specialised stuff from the medical side sometimes temporary hired by each hospital, also named data entries. The data is collected and introduced into the platform manually. The considerable volume of patients that Spanish hospitals have received since the onset of the pandemic limits the number of medical professionals available on non-assistencial tasks, which makes data collection a complex task and susceptible to errors.

Inside REDCap, data is structured in entries, such that each entry represents a different patient. An entry consists in a series of forms, which contain organised registers with blank fields to fill. Each blank usually represents a variable that is then stored. These forms do not have a linear structure, and there are fields that depend on other variables. For instance, there are variables that depend on another variable having a certain value in order to appear (*e.g.*, duration of a treatment only appearing if a treatment is applied). The dependency hierarchy of a certain form can be obtained as an HTML file, but it is very complex and the structure it provides is very difficult to navigate. To access the data, REDCap offers an API. REDCap API is limited and does not allow to get more than a 1,000 patients at the same time.

REDCap codifies variables as textual (numerical and textual values, and dates), radio (categorical variables, shown in the forms as a radio button) and checkboxes (categorical values with more than one possible choice at the same time). Radio variables are codified as integers, in which usually "Unknown" or missing values are codified as a numerical value too. For checkbox variables, one binary value is stored per possible choice (*e.g.*, a checkbox variable with 5 choices needs 5 binary values to store). These representations are not efficient in terms of storage, and easily allow datatype-related errors such as a data entry writing letters in what should be a numerical value (*e.g.*, "175 cm" instead of "175", "37°C" instead of "37").

Some of the variables may have more than one value at the same time, like bacterial or hemorrhagic complications suffered, or tests received. A patient may have suffered 3 different tests, performed at a different date each and with different results. In these kinds of situations, the fields in the form have dependencies with one another (*e.g.*, the fields for the second complication/test only appear if the first one is filled). In addition, each test or complication is stored in different predefined variables (*e.g.*, complication 1 variable, complication 2 variable, etc.), which is both storage-inefficient and a limited representation, as there is always a maximum amount of tests or complications that can be represented (for instance, only 3 bacteria complications can be defined).

Another case of dependency between variables are laboratory units. As different hospitals may use different units in their measurements, a form is dedicated for storing the units that are used in the measurements of entries. In the form corresponding to lab variables, for a certain variable, one value is stored for each possible lab unit. For instance, if haemoglobin was measured in g/dl or mg/dl, two values would be stored. A lab variable usually has 3 or 4 different unit options, and depending on the choice, one field or another appears in the form. Subsequently, one variable or another is the one that stores the value of the corresponding unit. This is inefficient in terms of space, as for a certain lab variable only a maximum of one variable, usually out of 3 or 4, stores a value.

In this project there is also a need to represent time-dependent data, as for instance blood analysis are performed periodically for a certain patient. As such, the forms of each entry are organised in events. Each event represents a different relevant timestamp (*e.g.*, hospital admission, admission in Intensive Care Unit (ICU), etc.). Each event has an associated date, and two or more of them may happen on the same date. When this happens, only one of the forms on the same date is filled and the others are empty except for a field that indicates that the information of the aforementioned forms is in a previous form.

3. Technical requirements and design

One of our first goals within the CIBERES-UCI-COVID project is to generate a database that could be used for any data analysis or AI purpose. Most of the data of each patient (*e.g.*, comorbidities, symptoms...) is present only once in REDCap, so the most natural way to translate this into a database is to design tables which contain a single data row per patient. There are exceptions such as blood analysis results, which are performed periodically. These need a table with several entries per patient and a time bound. As database model, we implement a relational database using MariaDB[2] which comes with a wide range of safety measures and it is faster and more efficient than MySQL[3]. Having the original data (and source of truth) as an external register like REDCap, it is fundamental to have a way of connecting to its API from our database and retrieve the data periodically (since the data is being added continuously during the project). It must be reliable, avoiding the corruption of data.

From clinical point of view, there is an interest of filtering patients according to different factors such as treatments received, events (*e.g.*, ICU admission and discharge, invasive mechanical ventilation start and end, etc), variables with value (*e.g.*, outcome or gender) and patient features (*e.g.*, hypertensive patients). For continuous variables, outliers are removed in order to avoid problems in statistical analyses. We were provided with a list of laboratory and ventilation variables and their normal ranges (on sick patients). For those observations that have variables outside its range, instead of removing the complete observation we ignore these variables.

In addition, results of analyses need to be presented in a clear and organised way such as reports. To avoid manually generating a large quantity of very similar reports, there is a need for an automatic report generator.

3.1. Integration with other data sources

Several studies about COVID-19 patients started concurrently with CIBERES-UCI-COVID. Due to the need for fast results, these studies worked independently and using uncoordinated data structures. Early in the project was decided to aggregate data from two other such studies into CIBERES-UCI-COVID because of the data volume they could offer, and the consequent benefits to any posterior analysis. Also, we establish to migrate these data into REDCap so that only a unique source of truth is maintained. These two independent studies are the SEMYCIUC consortium and Getafe hospitals, which both use spreadsheets as their main data storage. For each study an independent migration program is created, which maps the variables of the studies with the equivalent REDCap variables.

3.2. Target infrastructure

At this point, we define the dataflow of the project, in agreement with the identified technical requirements. In detail, we seek to have a single source of truth which can always be relied on, control over the data by those in charge of managing the analysis and enabling REDCap data modification with improvements made during analyses.

[2] MariaDB is an community-developed open source database system (https://mariadb.com/docs/)

[3] MySQL is an open-source database system developed by Oracle (https://dev.mysql.com/doc/)

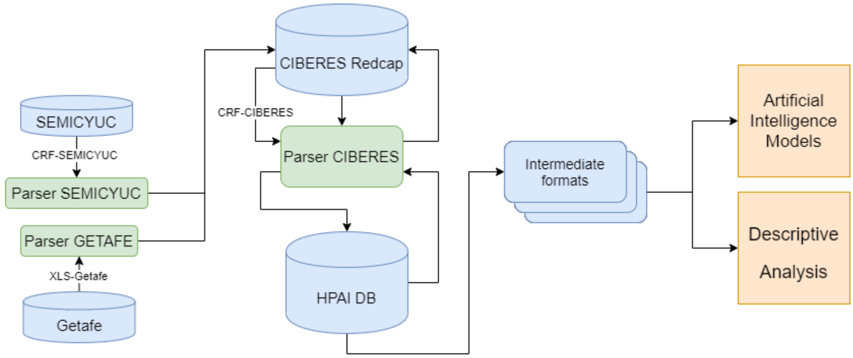


Figure 1. Dataflow implemented in CIBERES-UCI-COVID. As cylinders, data sources. In green, data parsers implemented by the authors.

This results in the scheme shown in Figure 1. The two external sources of data from SEMICYUC consortium and Getafe hospitals each have a specific parser (Parser SEMICYUC and Parser GETAFE), which makes them compatible for integration with REDCap. There is also a parser associated to REDCap's data (Parser CIBERES), which outputs an immutable database (HPAI DB) ready to be exploited by our algorithms.

4. Integration, reports and security

Beyond the technical requirements, we also identify a set of functionalities which are necessary to make a project of this scale feasible. For example, continuous development and integration (CI/CD), which allows us to validate if changes, after the code has been integrated in the app, are stable and correct. Currently this is a desirable feature since it allows to detect and repair failures faster and create workflows across the development, testing, and production environments. This process is automated by a pipeline, which contains all the stages needed to deliver a new version of the application. The pipeline itself can be divided in three sub processes: Code integrity and validation testing (including Unit tests and Integration tests), database migration (including error handling and data recovery) and automated generation of different types of reports.

4.1. Data migration

Data migrations from REDCap to the database are performed periodically, so that the database is kept updated when new registers are added to REDCap. The migration of REDCap data to our database consists of several sequential steps.

The forms that can be filled once per event are checked before the actual migration starts. The migration processes separates forms of the same type by date, and for each group of forms with equal date it looks for the one that is filled and duplicates the information in the rest. This way, our database explicitly contains all information. Because the retrieved data from REDCap is formatted entirely as text strings, the first step is casting each value to the corresponding type in our database. In a few exceptional cases, an additional string processing step is required to remove strange characters (e.g. temperature as “37^{o}C” instead of “37”). When possible, a direct mapping between a REDCap value and a table column is defined and the value is inserted in the column as is. However, some of the columns from the database require additional transformations, like unifying measurements in various units. For these cases, a function that transforms the original value is defined. Both the direct mapping and the needed functions are specified in a configuration JSON file. Due to the number of patients and the amount of data available for each of them, this process has been parallelised.

4.2. Report generation

Report generation is of special interest given their key role in the interaction with the medical participants. In this case, a library for automatically generating reports in different formats has been implemented. The system can generate table (csv files), correlation (matplotlib plots) and text reports (docx files). The latter is used to display results of the different analyses such as missing value and outlier analyses.

The process for generating a report is composed of three main steps. In the first place, the user instantiates a Configuration object. This class contains the minimum variables required to generate reports such as the list of variables and population filters but also contemplates the possibility of adding new fields easily. As a result, the system builds a validated dataframe with the requested variables and the restrictions already applied to it. The conducted analyses are applied to these data and their results are saved in the corresponding report.

Finally, considering the personal nature of all data being handled, special focus has to be put onto security measures. All data is stored in a private isolated – via virtualisation – server placed in the EU (Spain). Only an automated GitLab[4] CI/CD pipeline has access to this server unless emergency access is needed, and in this case only authorised researchers can access via SSH tunnelling using a private key. All private data is stored on a MariaDB database. If data has to be uploaded or downloaded by third parties, we use a MinIO[5] encrypted – at rest and in transport – distributed S3-compatible storage server. All credentials related to the project are stored in a secure Hashicorp vault[6]. Tokens to access this vault are only available to specific code repositories and authorised researchers. By combining repository-based authentication with a token-based vault, we enforce that the storage and processing of data is carried out, in an automated fashion, by code reviewed and pushed by the authorised researchers.

Communication with doctors as well as file sharing are done through GitLab and Rocket.Chat[7] with mandatory two-factor authentication. Role-based permission management is enabled in GitLab, so all users can participate in writing and commenting tasks but only a few users can see or push code. This allows us to restrict, track and audit who, when and why users access data, as well as to carry out a collaborative follow-up and evolution of the sub-studies.

[4] GitLab is a version control application with some DevOps tools https://docs.gitlab.com/

[5] MinIO is an encrypted cloud storage service (https://min.io/)

[6] Vault is a data protection and management tool (https://www.hashicorp.com/products/vault)

[7] Rocket.Chat is a chat service placed on a private server (https://docs.rocket.chat/)

5. Data pre-processing

In this section we review the main pre-processing steps performed, prior to any consequent analysis. Those are the detection and handling of missing values, the definition of outliers, the analysis of correlated data, the creation of new features, and the detection and management of errors.

Missing analysis and imputation One of the most common situations when working with real data is the existence of missing data. These missing values arise due to many reasons such as undefined values, data input errors, irrelevant information, mismatch of variables between databases, etc. In the context of CIBERES-UCI-COVID, where data is introduced by tens of different data entries (each hospital hiring its own), and where hundreds of medical variables are requested for each sample/patient, missing data is frequent and must be addressed thoroughly. Not handling missing data properly can have a negative impact on performance of machine learning models. As the authors of [6] point out, missing values can reduce statistical power and representativeness of samples, introduce bias and reducing drastically the quality of the study. In our case, we obtain a median of 168 (0.0065%) (IQR 12-9466) missing values per variable and 248 (0.0096%) (IQR 209-351) missing values per patient.

Missing values are replaced by estimated values based on other variables using K-Nearest-Neighbour or Multiple Imputations by Chained Equations techniques, using the fancyimpute library[8]. In view of those considerations, CIBERES-UCI-COVID offers a set of techniques that analyses missing data in depth in order to understand its nature and address the issues mentioned before. Depending on the study being carried and the variables being targeted by them, a combination of some or all of these methods can be applied, so we implemented them in a way that they can be activated, deactivated, and composed.

Outliers analysis Before performing any statistical analysis, outliers are to be removed. To identify outliers, medical expertise is of capital importance, as they can define the data ranges that can be considered as feasible. There are variables that must be carefully supervised due to their clinical importance. If outliers are located in large numbers, the project coordination must consider the possibility of contacting the hospital in question to understand and fix the source of outliers. For this reason, an analysis of outliers by variable and by hospital is carried out. Taking into account continuous variables, a total of 6468 (0.22%) outliers are generated with a median of 44 (IQR 26.5-144) outliers per variable.

Correlation analysis To interpret the relationships that may exist between features, a correlation analysis is performed. Understanding these relationships is useful in order to avoid multicollinearity and redundancy issues. Correlation is computed by different means depending on the data types. For pairs of continuous variables, Pearson is used to measure their statistical relation. For pairs of categorical variables, we measure the correlation ratio [2] by computing the relation between the statistical dispersion within individual categories and the dispersion across the whole population or sample. Finally, for pairs of categorical and continuous variables we use Cramer's V [1], which is a mea-

[8] fancyimpute is a python imputation library. We used version 0.5.4, this library was developed by Alex Rubinsteyn and Sergey Feldman (https://github.com/iskandr/fancyimpute)

sure of association based on Pearson's chi-squared statistic between two nominal variables (a value between 0 and 1). Correlations are computed for all pairs of features in a dataframe and reported to the medical team to decide which variables have higher clinical relevance.

Feature engineering After the collection and transformation of REDCap data, we compute derived medical variables and the enrichment of other ones through domain knowledge. It is the case of scoring systems such as APACHE II or SOFA used to measure the critical state of a patient. Most derived variables are computed and saved in the database after migration. There are others that are used only in very particular studies, so it is not necessary for them to be saved permanently. For instance, delta variables, which compare the results for a variable in different events, are computed at execution time. These variables are used to measure the evolution of a variable as the patient progresses through the different hospital stages.

Some variables are enriched with knowledge from medical experts. This includes variables that can change their value when some conditions are fulfilled. Usually, these are variables that are involved in the calculation of other more complex variables. This is the case for example of the Glasgow Coma Scale, if it equals 15 for some medical event then the rest of the events take this value.

Error handling The error handling is basically divided into two tasks: controlling the errors that may occur in our system and those derived from the REDCap data. For the former, as mentioned above, we perform unit and integration testing. This allows us to quickly detect and resolve bugs, refactor and improve the code, reduce complexity and ensure that all code meets quality standards before it is deployed.

In the case of REDCap errors, our task is to inform the centers so that they can be corrected in the platform. Among the errors found we distinguish variables with wrong units, with impossible (the value is very far from the normal range) or inconsistent values such as variables whose value may be incorrect when observed in combination with others, either because they are part of a sequence or that depend on other variables.

The risk that we encounter these types of errors is high when data is entered manually, so we must be careful when processing it before conducting any analysis.

Feature selection In order to be able to determine those variables that have a greater impact predicting the outcome of interest, importance rankings are generated using several feature selection techniques. As conventional methods we find regression (lasso, logistic) and classification algorithms (random forest) and embedded methods (recursive feature elimination). Techniques dedicated to survival analysis have also been incorporated extracted from PySurvival [9] library.

Statistical analysis It is performed to interpret data and discover patterns and trends. Given a set of variables, categorical variables are presented as frequency/percentage of a group from which they were derived, and for continuous variables the median [interquartile range (IQR)] is used. Categorial variables were compared with the use of Chi-square test or Fisher's exact test, while continuous variables were compared with the Student's t test or Mann–Whitney U test. For comparing variables between more than two groups one-way ANOVA and Kruskal tests are used. Missing values for each feature are ignored.

[9] pySurvival is an open source python package for Survival Analysis modeling (https://square.github.io/pysurvival/)

6. Enabling sub-studies

The scale of the CIBERES-UCI-COVID project (both in terms of samples and features) enables lots of research lines. These are called sub-studies, specialised research projects which feed off the project. In order to facilitate and scale sub-studies, we implement a set of enabling methodologies. In particular, a global configuration has been defined in order to avoid code repetition, automate pre-processing, report generation process and to generate validated data with the appropriate format for each sub-study.

The global configuration is a guide for the system to create the validated dataframe that includes the variables and the chosen population by the responsible of each sub-study. Once the dataframe is built, it is returned and its format is adapted to build the report. Among other things, the configuration contains information about the target type of report (*i.e.*, table, document or correlation report), list of variables needed and to be derived, how to split the population and variable ranges for imputation purposes.

The other main consideration for sub-studies is the population upon which it is based. This is defined by the values of certain variables (*e.g.*, age, gender, comorbidities, severity of disease, time of admission *etc.*), by mandatory variables (which cannot be missing) and by exclusion filters.

In addition to implementing the necessary tools to build a dataset to work with, CIBERES-UCI-COVID has developed a framework on which artificial intelligence techniques (classification, clustering, survival analysis techniques) can be easily incorporated. Thus, allowing the use of artificial intelligence to be within the reach of any clinical study under development. As a result, CIBERES-UCI-COVID has enabled the successful execution of several clinical studies such as [3,8,7] and others that are in progress. Currently under development are studies that aim to analyse the influence of intubation and tracheostomy strategies on the evolution of critical patients, risk factors involved in respiratory infection and the influence of metabolic disorders in the progression of the illness among others. Regarding survival, several methodologies have been implemented for predicting survival and analysing risk factors in hospital mortality.

7. Conclusion

In this paper, we describe the methodologies used in CIBERES-UCI-COVID consortium. One that spanned tens of hospitals, hundreds of variables and thousands of patients. These have allowed us to create a database with pre-processed, accessible and updated data from different data sources. With that in place, the project can foster multiple analysis and learning methods, carrying out studies of interest and high impact. In fact, roughly a year after the project started, several relevant research papers [3,8,7] have already been published thanks to this work, and to the data collected from the REDCap platform by means of regular migrations, and from the SEMYCIUC consortium and Getafe hospitals, through specialised parsers.

CIBERES-UCI-COVID also provides a set of pre-processing steps that allow to examine data in depth before conducting any further analysis. Among those steps we find missing analysis and imputation, outliers and correlation analysis in addition to feature engineering to enrich data after cleaning and error handling in a effective way. Results of those analysis are presented as reports through tables (csv files) and documents (docx files) generated in an automated way. Pre-processing and data retrieval are automated as well. A global configuration is used in order to manage which variables and filters we want to report and its format.

To sum up, this work has resulted in a scalable, efficient and flexible tool to generate validated data adapted to an arbitrary number of medical studies while collectively establishing and strictly following a set of good practices in design, implementation and security. Significantly, this work has been done under pandemic conditions, in less than a year, and under the scientific pressure generated within a huge project such as CIBERES-UCI-COVID.

Acknowledgements

The authors are very grateful to CIBERES-UCI-COVID consortium with special mention to Antoni Torres, Anna Motos, Laia Fernández-Barat, Ferran Barbé and Albert Gabarrús for their collaboration and support in the development of this paper.

References

[1] Harald Cramer. *Mathematical methods of statistics.* Princeton University Press, Princeton, 1946.

[2] R.A. Fisher. *Statistical methods for research workers.* Edinburgh Oliver & Boyd, 1925.

[3] Jessica González, Iván D Benítez, Paola Carmona, Sally Santisteve, Aida Monge, Anna Moncusí-Moix, Clara Gort-Paniello, Lucía Pinilla, Amara Carratalá, María Zuil, et al. Pulmonary function and radiologic features in survivors of critical covid-19: A 3-month prospective cohort. *Chest*, 2021.

[4] Paul A Harris, Robert Taylor, Brenda L Minor, Veida Elliott, Michelle Fernandez, Lindsay O'Neal, Laura McLeod, Giovanni Delacqua, Francesco Delacqua, Jacqueline Kirby, et al. The redcap consortium: Building an international community of software platform partners. *Journal of biomedical informatics*, 95:103208, 2019.

[5] Paul A Harris, Robert Taylor, Robert Thielke, Jonathon Payne, Nathaniel Gonzalez, and Jose G Conde. Research electronic data capture (redcap)—a metadata-driven methodology and workflow process for providing translational research informatics support. *Journal of biomedical informatics*, 42(2):377–381, 2009.

[6] Hyun Kang. The prevention and handling of the missing data. *Korean journal of anesthesiology*, 64:402–6, 05 2013.

[7] Lucía Pinilla, Ivan D Benitez, Jessica González, Gerard Torres, Ferran Barbé, and David de Gonzalo-Calvo. Peripheral blood micrornas and the covid-19 patient: methodological considerations, technical challenges and practice points. *RNA biology*, 18(5):688–695, 2021.

[8] Ana P Tedim, Raquel Almansa, Marta Domínguez-Gil, Milagros González-Rivera, Dariela Micheloud, Pablo Ryan, Raúl Méndez, Natalia Blanca-López, Felipe Pérez-García, Elena Bustamante, et al. Comparison of real-time and droplet digital pcr to detect and quantify sars-cov-2 rna in plasma. *European journal of clinical investigation*, page e13501, 2021.

[9] Antoni Torres, María Arguimbau, Jesús Bermejo-Martín, Raquel Campo, Adrian Ceccato, Laia Fernandez-Barat, Ricard Ferrer, Natalia Jarillo, Jose Ángel Lorente-Balanza, Rosario Menéndez, et al. Ciberesucicovid: A strategic project for a better understanding and clinical management of covid-19 in critical patients. *Archivos de Bronconeumologia*, 2020.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210143

Cutting Tool Wearing Identification Through Predictive Maintenance and Its Impact on Surface Quality

Jose Maria GONZALEZ CASTRO[a,1], Giselle RAMIREZ SANDOVAL[b],
Eduard VIDALES COCA[b], Nuri CUADRADO LAFOZ[b], Francesc BONADA[a],
[a] *Eurecat, Centre Tecnològic de Catalunya, Unit of Applied Artificial Intelligence, Av. Universitat Autònoma, 23, 08290 Cerdanyola del Vallès, Spain*
[b] *Eurecat, Centre Tecnològic de Catalunya, Unit of Metallic and Ceramic Materials, Plaça de la Ciència 2, 08243, Manresa, Spain*

Abstract. Smart manufacturing has been in the media for a long time, but the reality shows that traditional mechanical manufacturing industries have not been able to implement data solutions aligned with Industry 4.0 standards. This work inquiries into the possibility of measuring cutting tool vibrations for CNC turning machines and presents the data analysis and a predictive model to identify tool wearing that can affects integrity surface quality of the manufactured component. These preliminary results are orientated towards implementing a predictive maintenance methodology in cutting tools.

Keywords. CNC turning machine, Machine Learning, Tool Wearing Prediction, Industry 4.0

1. Introduction

During last decades there has been a special attention in predicting surface quality through vibration measurement during the milling process. This has been carried out to introduce Industry 4.0 technologies into metal manufacturing workshops where data-driven models are not used due to the lack of measured data during manufacturing. In this field, interesting research has been devoted to develop surface quality prediction methods for the roughness quality [1] [2]. Nevertheless, these works have been developed on a controlled lab where manufacturing parameters are known and recorded. Although the research shows promising results, real production applications face with increased challenges, such as technicians availability to introducing manufacturing parameters in an external software during process planning. Industrial manufacturing has to deal with dynamic scenarios that affect parameters such as tool wearing, changes in the material batches, also the technician can modify machining parameters as feed rate or the radian and axial depth cut during manufacturing [3]. These complex scenarios present difficulties for the implementation of previously developed models for on-line surface quality prediction, especially for small workshops.

[1] Jose Maria Gonzalez Castro, Eurecat, Centre Tecnològic de Catalunya, Applied Artificial Intelligence Unit, Av. Universitat Autònoma, 23, 08290 Cerdanyola del Vallès, Spain; E-mail: jose.gonzalez@eurecat.org.

The objective of the research presented in this work is to develop a predictive model that can identify surface quality degradation during manufacturing, reducing the costs of manufacturing defective components and identifying the problem when quality assurance tests are performed. AVINT project pushes forward the current surface quality prediction techniques by researching into the possibility to develop a model that only depends on on-line measured vibration data that can be correlated with tool wearing. This approach will allow to shift from current preventive maintenance strategies to change the cutting tools into a more effective and cost-saving predictive maintenance strategy where tools are used until they really need to be changed, maximizing both tool lifespan and part quality.

2. Experimental Data and Feature Engineering

The data schema consists of two sensors registering 4 channels. The first three channels correspond to the 3 axes from an accelerometer with a sampling frequency of 10 kHz. The fourth channel corresponds to an electro-acoustic sensor with a maximum sampling frequency of 200 kHz. These 4 channels are stored on a time-series database and organized by experiment.

A set of experiments were performed on a HAAS ST-10Y CNC lathe machine before performing industrial measurements on a real production facility. Five tools with different levels of degradation were tested. The objective was to evaluate the possibility to determine the tool degradation based only on recording the vibrations produced during operation.

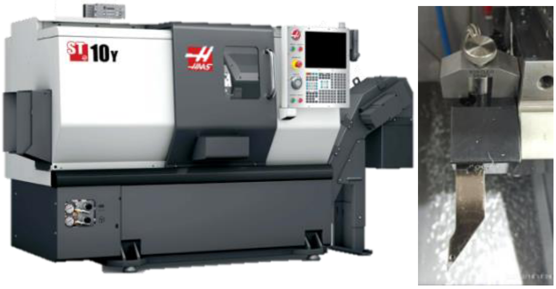


Figure 1. experimental setup

Five different tools were tested by manufacturing a cylindrical test specimen of aluminium alloy. Three cylindrical specimens were manufactured for each tool. The tools were named as e1, e2, e3 e4 and e5. Tools e1 and e3 were new tools, while e2, e4 and e5 had different wear level at the cutting edge produced under lab control before the machining test (see Table 1). Worn material of tools was measured by means of a focus variation microscope obtaining 3D images of the cutting edges.

Table 1. Wear at the cutting edge before machining test

	e1	e2	e3	e4	e5
Removed material [mm3]	New	0.017	New	0.001	0.019

The data has been segmented to consider only periods where the selected tool is working, to obtain a region of interest to be analyzed. To extract important information from the data and to reduce the dimensionality, statistical descriptors were extracted for each channel, considering both time and frequency domains and enabling to reduce the sampling values to 0.1 Hz. The time-domain descriptors consist of the mean value, standard deviation, skewness, kurtosis, maximum peak value and the dispersion of the 10 most prominent peaks on each 0.1 seconds of data. Frequency data consisted of the energy of the signal for different frequency bands based on the work described in [4].

3. Analysis of the Experiments

Due to the high dimensionality of the data descriptors, a manifold Multi-Dimensional Scaling (MDS) is used to visualize the data in a two-dimensional space area. MDS establish the possibility to distinguish between tools only using the data by representing the similarity contained in the data as a geometrical distance. Figure 2 shows the MDS visualization results.

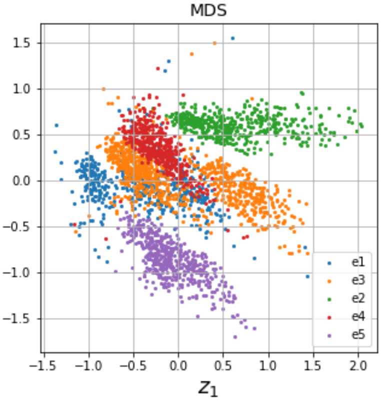


Figure 2. Result of visualizing vibration data on two dimensions through Multi-dimensional scaling (MDS).

The MDS presented in Figure 2 shows that tool set e1, e3 and e4 contain a higher level of similarity with respect to other tools, mainly expressed in the formation of a single cluster. On the other side, tools e2 and e5 show their data clustered on opposite sides from the main cluster showing dissimilarity with all the other tools.

The surface quality of the manufactured component was assessed by measuring four roughness parameters (see Table 2): the arithmetical mean heigh for line (Ra) and surface (Sa), the mean peak width/average length of elements (Rsm) and the developed interfacial area ratio (Sdr)[5]. These values were measured by means of confocal microscopy technique using a Sensofar software and according to ISO 4287 and ISO 25178.

Table 2. Surface quality of the manufactured cylindrical specimens

Tool	Ra [µm]	Sa [µm]	Rsm [mm]	Sdr [%]
e1	6.95±0.34	6.97±0.23	0.20±0.01	8.17±1.53
e2	11.87±2.27	13.63±1.68	0.22±0.01	62.30±4.10
e3	7.36±0.38	7.33±0.10	0.19±0.01	10.65±1.07
e4	8.97±0.71	8.97±0.27	0.19±0.01	17.60±0.30
e5	4.09±0.79	4.33±0.89	0.25±0.01	2.80±0.60

Cutting edge of tools after manufacturing tests are shown in Figure 3. The green color shows the initial geometry, red and blue colors represent the adhered and removed material, respectively.

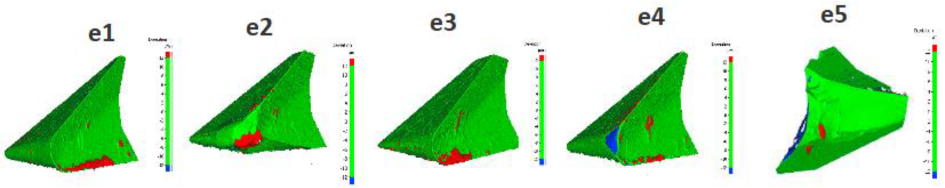


Figure 3. Cutting edge of the tools after test (images acquired by focus variation microscopy). The red color represents adhered material, blue color is the removed material from the tool, and the green color is the initial geometry of the tool. All the images are oriented to show the maximum variation of the tool with respect to the new tool. The images are scaled within -12 and +12 micrometers.

4. Discussion and Conclusions

Data shows that it is possible to distinguish between tools showing a similarity in the recorded data related to the surface quality and their degradation. More precisely, e1 and e3 being new tools, together with e4 which was slightly degraded produced surfaces with similar quality.

On the other side, e2 and e5 produced divergent signals with respect to other tools. This difference is seen on the removed material which is 10 times higher with respect to e4. The sharper cutting edge of e5 tool produced by microchipping before the manufacturing test is remarkable and it probably explains the small rugosity values generated during the experiment compared with other tools.

The analysis of the experiments shows an encouraging result that it is possible to track tool wear by only monitoring tool vibrations. The project is continuing by measuring tool vibration on real production environments. Currently, industrial data is available and further analysis are going on. The nature of the problem suggests that an unsupervised method can be used to identify on-line tool degradation by means of signal variation during cutting work.

5. Acknowledgment

This work has been carried out in the framework of the AVINT project, which has received funding from the European Union through European funding for regional development (FEDER).

References

[1] D. Korzeniewski y N. Znojkiewicz, «The influence of vibrations on surface roughness formed during precision boring,» Archives of Mechanical Technology and Materials, vol. 37, 2017.

[2] S. Palani y U. Natarajan, «Prediction of surface roughness in CNC end milling,» International Journal of Advanced Manufacturing Technology, pp. 1033-1042, 11 June 2011.

[3] P. Benardos y G.-C. Vosniakos, «Predicting Surface Roughness in Machining: a Review,» International Journal of Machine Tools and Manufacture, vol. 43, pp. 833-844, 2003.

[4] G. Quintana, M. Garcia-Romeu y J. Ciurana, «Surface roughness monitoring application based on artificial neural networks for ball-end milling operations,» Journal of Intelligent Manufacturing, pp. 607-611, 2011.

[5] KEYENCE, «Introduction to "Roughness",» [Online]. Available: https://www.keyence.eu/ss/products/microscope/roughness/. [Last acces: 10 05 2021].

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210144

Mental Workload Detection Based on EEG Analysis

José YAURI [a,1], Aura HERNÁNDEZ-SABATÉ [a], Pau FOLCH [b] and Débora GIL [a,c]
[a] *Computer Vision Center, Universitat Autònoma de Barcelona, Spain*
[b] *Aslogic, Barcelona, Spain*
[c] *Serra Hunter Fellow*

Abstract. The study of mental workload becomes essential for human work efficiency, health conditions and to avoid accidents, since workload compromises both performance and awareness. Although workload has been widely studied using several physiological measures, minimising the sensor network as much as possible remains both a challenge and a requirement.

Electroencephalogram (EEG) signals have shown a high correlation to specific cognitive and mental states like workload. However, there is not enough evidence in the literature to validate how well models generalize in case of new subjects performing tasks of a workload similar to the ones included during model's training.

In this paper we propose a binary neural network to classify EEG features across different mental workloads. Two workloads, low and medium, are induced using two variants of the N-Back Test. The proposed model was validated in a dataset collected from 16 subjects and shown a high level of generalization capability: model reported an average recall of 81.81% in a leave-one-out subject evaluation.

Keywords. Cognitive states, Mental workload, EEG analysis, Neural Networks.

1. Introduction

A cognitive state is the state of the mind, often named cognitive status, and it is related with the human performance and awareness in a specific time. Usually, cognitive states of workload, distraction, and fatigue are among the most studied due to their association to human performance and reliability and their risks for catastrophic effects in, for instance, aviation and automotive accidents [1,2,3].

In particular, if we define mental workload as the cognitive and psychological effort to conclude a task [4] we can observe that when workload is too heavy or too light it can degrade the human performance [5,6]. Mental workloads have high effects on the daily life human performance, since the more difficult the task is, the greater the mental workload [7] results. Thus, the study of workload becomes essential to prevent accidents, since it could compromise human task performance [8].

Since workload involves cognitive, neuro-physiologic, and perceptual processes to resolve a task, it is affected by individual capabilities, motivation, as well as, its physical

[1] Corresponding Author: Campus UAB, Edifici O, s/n, 08193 Cerdanyola del Vallès, B, Barcelona, Spain; E-mail: jyauri@cvc.uab.cat

and emotional state [9]. Although this multifaceted nature of workload prevents to study workload directly, it is feasible to be inferred from different quantifiable variables [10]. There exist many proposals for recognizing workload based on physiological features, such as hearth rate, eye movement and dilation, electroencephalogram (EEG) and electrocardiogram (ECG) [11,12].

Besides, the recent emerging of low cost EEG headsets has driven them to new researches (like interaction with home devices, teaching-learning educative methods or mentally control robotic arms) further than medical screening of neurological disorders. In the particular case of cognitive state assessment, EEG alone is becoming the preferred sensor for addressing its characterization [13,14,15]. However, there is not enough evidence in the literature to validate how well models generalize in case of new subjects performing tasks of a workload similar to the ones included during model's training.

In this paper we propose the use of EEG for characterizing workload by means of a neural network and show its ability to generalize the model across a wider population.

The remainder of this paper is organized as follows: Section 2 presents relevant related works, Section 3 explains the process followed to collect the data. Section 4 presents our proposal for the analysis of EEG generalization capabilities, while Section 5 is devoted to present the results of our experiments. Finally, Section 6 outlines the conclusion and future work.

2. Related work

The most generalized mechanisms to measure workload can be split in two main categories [7,11,1]: subjective measures based on the subject perception and objective scores based on physiological responses.

On the one hand, subjective measures are still the most used to assess mental workload, being the NASA Task Load Index (TLX) [16] the most prominent test to gain insight about the perceived workload levels while a subject works with various human-machine interface systems [4,17]. This questionnaire measures the mental workload based on a weighted average of six sub-variables: mental demand, physical demand, temporal demand, performance, effort and frustration and it is widely used in aviation to assess mental workload of pilots while interacting with plane controls [18,19].

On the other hand, physiological measures provide a more reliable data of workload by measuring physiological dynamic changes which cannot be controlled consciously, so they are becoming more popular among researchers in recent years [20,21,22]. The most common sensors to record physiological data are: electrocardiogram (ECG) to register heart's electrical activity, electromyograph to read skeletal muscles electrical activity, electroencephalogram (EEG) to detect electrical activity in the brain, photoplethysmography to register volumetric changes in the blood flow, respiration rate sensors, electrodermal activity (EDA) to read skin surface temperature, oxygen density in the blood in the brain, and eye movement trackers, among others [23]. TLX surveys allow to assess the perceived workload [16], but it is highly subjective. However, physiological data occurs spontaneously, and, together with TLXs, provide a more reliable information [20,11,4].

The combination of several physiological sensors to classify workload states gives better results than using a single one. The approach proposed in [24] combines EEG, ECG, and electrooculography (EOG) and results show a highest predictive power for their combination (80%) rather than the analysis of each one independently (70%). Besides, the study in [12] reports an accuracy average of 85.2 (± 4.3%) combining EEG, ECG, respiration rate, and EDA to classify 4 mental states. The work in [25] still shows better results combining EEG, ECG and EDA than using only EEG signal from classifying four mental states, although results from the single sensor are promising (86.66%).

At that point, Deep Learning (DL) approaches are gaining ground over more classical machine learning techniques due to their ability to automatically extract the features [23,26]. For instance, the study in [8] proposes a concatenated structure of deep recurrent and 3D convolutional neural networks to combine both raw and spectral EEG data and assess two degree of mental workloads reporting an average accuracy of 88.9% in a cross-task assessment.

However, none of the last previous works were tested on a dataset totally unseen in the training set, being their ability to generalize an unknown.

In this work, we propose to investigate the ability of 1D-CNN models to recognize two levels of mental workload from EEG signals and generalize the model to an unseen population in the training set. To induce low and medium workload, we propose several modifications of the N-back test [27] to collect data from subjects. Regarding workload classification, we use a neural network ensemble (NN) trained on the power spectrum of filtered EEG theta waves. To assess the generalization abilities of models we propose a personalized model trained for each individual and a generalist one trained on the whole data set. Results obtained in a dataset collected from 16 subjects show a high level of generalization capability with average recall of 81.81% in a leave-one-out subject evaluation.

3. Data Set Collection

When performing different tasks along the day, people experiment different levels of mental workload depending on the level of attention required, the difficulty of such task and how many sub-tasks are needed to take care off. In order to induce different levels of workload in a controlled manner, we propose to use N-Back-tests [27].

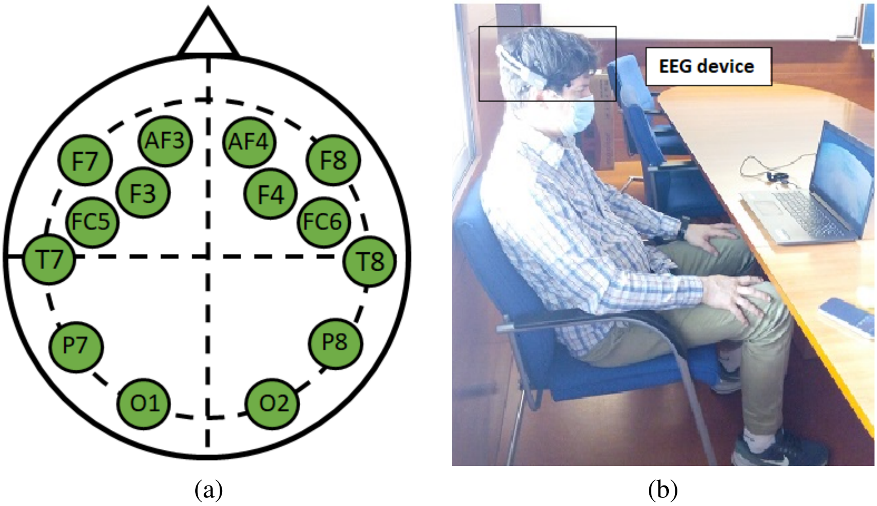
N-Back-tests are memory demanding games requiring the resolution of tasks according to a stimulus presented N trials before. We used three variants of the N-Back-tests to induce low, medium, and high mental workload:

1. *Position 1-back for low workload.* A square appears every few seconds in one of eight different positions on a regular grid over the screen. Players must press a keyboard key in case the position of the square on the current screen is the same as the square of the previous grid.
2. *Arithmetic 1-back for medium workload.* An integer number between 0 and 9 appears every few seconds on the screen while an audio message says an arithmetic operation (plus, minus, times and divide). Players have to solve this operation using the current number and number that appeared in the previous screen.
3. *Dual arithmetic 2-back for high workload.* This test combines the two previous ones. An integer number between 0 and 9 appears every few seconds in one of eight different positions on a regular grid. At the same time, for each number that appears on screen, an operator is presented with an audio message. As before,

players have to solve this operation using the current number and number that appeared in two screens before. In addition, players have to press a key in case the position of the current number is the same as the position of the number shown two screens before.

The neurophysiological response of a subject against mental demanding tasks depends on its baseline state, which is prone to vary across time. In order to account for differences in the baseline state of subjects, previous to the N-back-tests participants watched a relaxing video for 10 minutes. For each experiment (low, medium and high workload), we call the video watching, phase 1, and the N-back-test, phase 2. After the game, participants ask a TLX questionnaire to collect their subjective perception of game difficulty and workload.

Although, this work presents results on EEG, we also recorded the electrocardiogram (ECG) data during the video watching and the game. For EEG recording, we used the EMOTIV EPOC+ headset [28] which has 14 electrodes placed according to the 10/20 system. This sensor provides both raw data and power spectrum for the main brain rhythms (theta, alpha, beta low, beta high, and), at 128 Hz and 8 Hz, respectively. Figure 1 illustrates the distributions of electrodes of this sensor (a) and a volunteer during a session task (b).



(a) (b)

Figure 1. Data collection with Emotiv Epoc+ headset. (a) Electrodes distribution over the head scalp. (b) A volunteer during a N-Backtest.

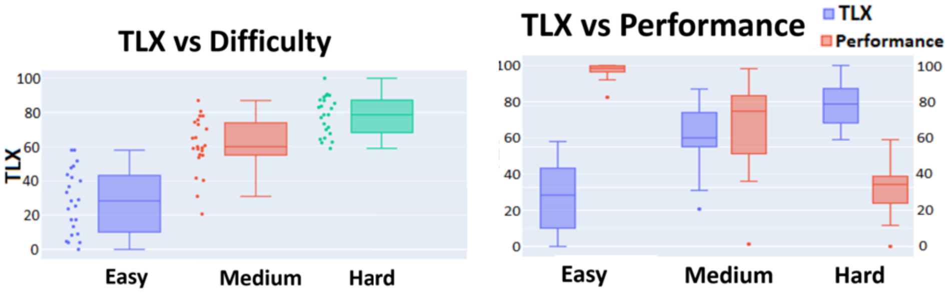
A total of 24 subjects participated in the experiment. Subjects were adults between 20 and 60 years, all of them were healthy without any condition that might have cause an imbalance in the data recorded. The sequence of tasks were randomly assigned to subjects, and recording of each session was in different days and hours.

4. Machine learning approaches

In order to assess to what extend a general model trained over a set of individuals can successfully predict a new unseen individual, we have considered the following approaches for the analysis of EEG generalization capabilities:

- **Personalized model for each individual**. A different model is trained for each subject of the data set to account and compensate for large intra subject variability in EEG signals.
- **Generalist model for the population**. A single model using all subjects is trained to assess whether inter subject variability can be properly modelled.

For both approaches we implemented a binary neural network trained to classify between workload (phase 2) and base lines (phase 1) phases. The experiment used to define the training data of the WL class was the phase 2 of the second experiment (noted as WL2). The phase 2 of the first experiment was discarded as training data because, according to TLX, it did not demand any significant mental effort for most users (Figure 2.a). The phase 2 of the third experiment was also discarded as training data because, according to TLX and users' performance, most users considered the task too difficult and gave up at some point of the experiment (Figure 2.b). Regarding the baseline class, all phases 1 can be considered for training. This data will be noted BLi, i=1,2,3, for i indicating the experiment. In order to discard any dependency of models with respect base line acquisition, we trained 3 different models for each approach using a different BLi for the base line class: WL2 vs BL1, WL2 vs BL2 and WL2 vs BL3. Additionally, for the generalist approach we trained an extra model using all 3 baselines phases in an attempt to account for any variability across them and better model the space of the baseline class. This model will be noted as BLall vs WL2.



(a) (b)

Figure 2. TLX-based subjective perceptions. (a) Perceived difficulty of tasks. (b) Achieved performance on tasks.

Given that proposed N-back tasks are memory demanding stressing games and base line phases consist in watching a relaxing video, the theta wave [29] is the best candidate for discriminating the different mental loads of our experimental phases. In this work, we use the power spectrum of theta wave (4–8 Hz) sampled at 8 Hz.

Eye blinking and sudden head movements introduce abrupt sharp peaks of large amplitude in the power spectra wave that should be filtered before using them as predictors of a mental state [20].

In particular, we use an Inter Quartile Range (IQR) [30] filtering strategy to detect outlier values associated to muscular movement wave peaks. Our IQR filtering is based on setting the value of the 99% percentile of the distribution to all points above it.

To ensure a high quality of signals, we further filter data according to the quality of the EEG during recordings provided by the headset itself. For each sensor and recorded sample, Emotiv reports the quality of the recording in a discrete scale with values in 0,1,2,3,4 indicating how good the contact between sensor and head is: 4 for optimum; 3 for good; 2 for medium; 1 for bad; 0 for none. For the sake of data with the highest possible quality while keeping a reasonable sample size signals with a 25% of bad recordings are discarded (< 3). Further, since there is no evidence about what are the most discriminative sensors that best correlate to the detection of mental workload, the whole phase is discarded if the signal of two or more of the sensors has a low quality. Finally, a subject is discarded if either all its base line or its workload phases are discarded, since, in this case, there is not enough data to define the binary classification. After this quality filtering, only 16 of the 25 subjects were selected for models training and testing.

In order to feed data to models, θ signals were cut in temporal windows of 5 seconds without overlap [12]. So the input data of the networks are the concatenation of the 5 second windows for the 14 EEG sensors ($14 * 40 = 560$-dimensional feature space). In order to account for the difference in units and magnitudes, input data was standardized using the mean and standard deviation of the training set.

Table 1 shows the architecture of the proposed neural network and its chosen network parameters after cross validation. For training, the NN used a batch size of 128, the weighted cross-entropy loss to compensate unbalances between baseline and workload phases, Adam [31] as optimization method, and reported the best results at 100 epochs with a learning rate of 0.0001.

Table 1. The proposed neural network architecture.

Layer type	Input size	Hidden unit	Parameters
Linear	560	128	71,808
Dropout (0.1)	128	-	-
ReLU	128	-	-
Linear	128	2	258
SoftMax	2	2	-

5. Results

The performance of the different approaches for detection of mental workload has been assessed using the accuracy (or sensitivity) for each class. Sensitivity measures the capability of the system to detect BL and WL classes. Since we have a binary classification problem with WL the positive class, then the sensitivity for BL corresponds to the specificity of the model.

In order to validate the reproducibility of each model, the following experiments have been conducted:

1. **Model Personalized for each Individual**. Reproducibility of personalized models has been assessed at intra-experiment level. For each model trained with a different base line, WL2 vs BLi, i=1,2,3, 10% of the samples were randomly chosen for testing the capability of discriminating workload at different times of the task. A high accuracy would proof that the variability of the EEG is stable and low while continuously repeating the same task.
2. **Generalist Population Model**. The validation of the capability for modelling a population was tested using a leave-one-out scheme for the 4 models considered: the 3 trained using a single baseline, BLi vs WL2, i=1,2,3, and the one trained using all 3 baselines phases, BLall vs WL2.

Table 2 summarizes the recall of baselines (BL) and workload (WL2) for the intra-experiment reproducibility. We report the 95% confidence interval for each class computed for all subjects (for each subject the average of BLi vs WL2, i=1,2,3 is computed). The overall recall for both classes is above 90%, which shows that workload and base line signals are different regardless of the time the experiment was conducted. However, the variability is large, which might be attributed to a suboptimal size of the temporal windows and the variability in mental effort across the task.

Table 2. Personalized model. Intra-experiment Reproducibility.

BL	**WL2**
92.8 ± 7.03	91.17 ± 5.35

Table 3 summarizes the recalls of baselines (BL) and workload (WL2) for the generalist model trained using a single BL and the aggregation of the three. The model trained aggregating the 3 baselines has a higher performance in detecting baseline states. According to a Student t-test of paired data this difference is significant (p-val= 0.0054) with an average improvement range of (-17.2522, -3.5812). Regarding detection of workload phases, both approaches perform similarly (p.val= 0.7159). For both approaches, there are 3 outliers in WL detection rate that, given the small sample size, are highly influential. If we remove them, we have that for the remaining 80% of the subjects, the average detection of idle and work load stages for the model that aggregates all BLs for training is, respectively, 76.08% and 73.23%. This suggests that the variability and nonstationarity that psychophysiological data exhibits could be modelled if enough data from subjects was gathered.

Table 3. Generalist Model.

	Model trained with single BL		**Model trained with all BLs**	
	BL	WL2	BL	WL2
All population	66.57 ± 13.11	65.42 ± 25.59	78.06 ± 10.75	65.00 ± 24.90
80% of population	66.10 ± 13.87	73.95 ± 18.62	76.08 ± 10.87	73.23 ± 19.07

6. Conclusions

The first experiment (Table 2) shows that work load and base lines signals are different regardless of the time the experiment was conducted. However, the large variability in accuracies indicates that the temporal window might be suboptimal and should be adapted to the variant mental effort across a given task. Results of the generalist model show that the variability in baseline cognitive states can be modelled provided that enough training data is available. This is supported by the higher performance of models aggregating all baselines.

The analysis of the results suggests the following improvements. A delicate issue that has an impact in the performance of methods is the filtering of signals required to remove muscular motion peaks and other artefacts. EEG pre-processing approaches have not been standardized, and even small changes in artefact removal strategy may result in differences with large effects on particular portions of the signal. In this study, we have adopted a filtering approach based on signal probabilistic distribution for outlier removal in the temporal space. We consider that muscular motion could be filtered calibrating muscular signals before test recording to set either the values or the frequency ranges associated to muscular motion. In this context, a classifier based on Fourier features will be further investigated.

Also the size of the temporal window might be a critical issue in order to properly include workload peaks. We have use 5 seconds windows following (Han, 2020), but recent authors suggest to use longer windows to capture EEG non stationary nature. The optimal window size should be further investigated.

In a future work, we have to ensure the availability of more data to achieve convergence without overfitting and to train more complex architectures. Recent architectures like convolutional/LSTM and Lambda Nets including attention modelling will be also studied.

Acknowledgements

This research is partially supported by the H2020 Clean Sky 2 project: "E-PILOT: Evolution of cockPIt operations Levering on cOgnitive compuTing Services" (Grant Agreement No. 831993), Spanish projects RTI2018-095209-B-C21 and Generalitat de Catalunya, 2017-SGR-1597, 2017-SGR-1624 and CERCA-Programme. DGil is a Serra Hunter Fellow.

References

[1] da Silva FP. Mental Workload, Task Demand and Driving Performance: What Relation? Procedia - Social and Behavioral Sciences. 2014 dec;162:310-9.

[2] Wickens CD. Situation awareness and workload in aviation. Current Directions in Psychological Science. 2002 aug;11(4):128-33. Available from: https://journals.sagepub.com/doi/10.1111/1467-8721.00184.

[3] Loft S, Sanderson P, Neal A, Mooij M. Modeling and predicting mental workload in en route air traffic control: Critical review and broader implications. Human factors. 2007;49(3):376-99.

[4] Borghini G, Astolfi L, Vecchiato G, Mattia D, Babiloni F. Measuring neurophysiological signals in aircraft pilots and car drivers for the assessment of mental workload, fatigue and drowsiness. Neuroscience & Biobehavioral Reviews. 2014;44:58-75.

[5] Proctor RW, Van Zandt T. Human factors in simple and complex systems. CRC press; 2018.

[6] Shaw JB, Weekley JA. The effects of objective work-load variations of psychological strain and post-work-load performance. Journal of Management. 1985;11(1):87-98.

[7] Averty P, Collet C, Dittmar A, Athènes S, Vernet-Maury E. Mental workload in air traffic control: an index constructed from field tests. Aviation, space, and environmental medicine. 2004;75(4):333-41.

[8] Zhang P, Wang X, Zhang W, Chen J. Learning Spatial-Spectral-Temporal EEG Features With Recurrent 3D Convolutional Neural Networks for Cross-Task Mental Workload Assessment. IEEE Transactions on Neural Systems and Rehabilitation Engineering. 2019 jan;27(1):31-42. Available from: http://www.ncbi.nlm.nih.gov/pubmed/30507536.

[9] Li D, Wang X, Menassa CC, Kamat VR. Understanding the impact of building thermal environments on occupants' comfort and mental workload demand through human physiological sensing. In: Start-Up Creation. Elsevier; 2020. p. 291-341.

[10] Hendy KC, Liao J, Milgram P. Combining time and intensity effects in assessing operator information-processing load. Human Factors. 1997;39(1):30-47.

[11] Heine T, Lenis G, Reichensperger P, Beran T, Doessel O, Deml B. Electrocardiographic features for the measurement of drivers' mental workload. Applied ergonomics. 2017;61:31-43.

[12] Han SY, Kwak NS, Oh T, Lee SW. Classification of pilots' mental states using a multimodal deep learning network. Biocybernetics and Biomedical Engineering. 2020;40(1):324-36.

[13] Zhang P, Wang X, Chen J, You W, Zhang W. Spectral and Temporal Feature Learning with Two-Stream Neural Networks for Mental Workload Assessment. IEEE Transactions on Neural Systems and Rehabilitation Engineering. 2019 jun;27(6):1149-59. Available from: https://pubmed.ncbi.nlm.nih.gov/31034417/.

[14] Lee DH, Jeong JH, Kim K, Yu BW, Lee SW. Continuous EEG Decoding of Pilots' Mental States Using Multiple Feature Block-Based Convolutional Neural Network. IEEE Access. 2020;8:121929-41.

[15] Wu EQ, Peng X, Zhang CZ, Lin J, Sheng RS. Pilots' fatigue status recognition using deep contractive autoencoder network. IEEE Transactions on Instrumentation and Measurement. 2019;68(10):3907-19.

[16] Hart SG. NASA-task load index (NASA-TLX); 20 years later. In: Proceedings of the human factors and ergonomics society annual meeting. vol. 50. Sage publications Sage CA: Los Angeles, CA; 2006. p. 904-8.

[17] Index L. Results of empirical and theoretical research. Advances in. 1990.

[18] Wickens CD. Situation awareness and workload in aviation. Current directions in psychological science. 2002;11(4):128-33.

[19] Parasuraman R, Sheridan TB, Wickens CD. Situation awareness, mental workload, and trust in automation: Viable, empirically supported cognitive engineering constructs. Journal of cognitive engineering and decision making. 2008;2(2):140-60.

[20] Wang Z, Yang L, Ding J. Application of heart rate variability in evaluation of mental workload. Chinese journal of industrial hygiene and occupational diseases. 2005;23(3):182-4.

[21] Stanton N, Salmon PM, Rafferty LA. Human factors methods: a practical guide for engineering and design. Ashgate Publishing, Ltd.; 2013.

[22] Jang EH, Park BJ, Kim SH, Chung MA, Park MS, Sohn JH. Classification of human emotions from physiological signals using machine learning algorithms. In: Proc. Sixth Int'l Conf. Advances Computer-Human Interactions (ACHI 2013), Nice, France. Citeseer; 2013. p. 395-400.

[23] Deep Learning in Physiological Signal Data: A Survey. Sensors. 2020 feb;20(4):969. Available from: https://www.mdpi.com/1424-8220/20/4/969.

[24] Ziegler MD, Russell BA, Kraft AE, Krein M, Russo J, Casebeer WD. Computational Models for Near-real-time Performance Predictions Based on Physiological Measures of Workload. In: Neuroergonomics. Elsevier; 2019. p. 117-20.

[25] Secerbegovic A, Ibric S, Nisic J, Suljanovic N, Mujcic A. Mental workload vs. stress differentiation using single-channel EEG. In: CMBEBIH 2017. Springer; 2017. p. 511-5.

[26] Ismail Fawaz H, Forestier G, Weber J, Idoumghar L, Muller PA. Deep Learning for Time Series Classification: A Review. Data Mining and Knowledge Discovery. 2019 jul;33(4):917-63.

[27] Jaeggi SM, Buschkuehl M, Jonides J, Perrig WJ. Improving fluid intelligence with training on working memory. Proceedings of the National Academy of Sciences of the United States of America. 2008 may;105(19):6829-33. Available from: www.pnas.orgcgidoi10.1073pnas.0801268105.

[28] Emotiv. EMOTIV EPOC+ 14-Channel Wireless EEG Headset;. Available from: https://www.emotiv.com/emotivpro/.

[29] Addante RJ, Watrous AJ, Yonelinas AP, Ekstrom AD, Ranganath C. Prestimulus Theta Activity Predicts Correct Source Memory Retrieval. Proceedings of the National Academy of Sciences of the United States of America. 2011 jun;108(26):10702-7. Available from: https://www.pnas.org/content/108/26/10702https://www.pnas.org/content/108/26/10702.abstract.

[30] Wasserman L. All of statistics : a concise course in statistical inference. Springer; 2010.

[31] Kingma DP, Ba J. Adam: A Method for Stochastic Optimization. 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings. 2014 dec. Available from: https://arxiv.org/abs/1412.6980v9.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210145

Alternation Measures for the Evaluation of Selfish Agents' Turn-Taking

Nikolaos Al. PAPADOPOULOS[a] and Marti SANCHEZ-FIBLA[b,1]
[a] *Department of Applied Informatics, University of Macedonia, Greece*
[b] *Department of Technology, Universitat de Pompeu Fabra, Spain*

Abstract. Multi-Agent Reinforcement Learning reductionist simulations can provide a spectrum of opportunities towards the modeling and understanding of complex social phenomena such as common-pool appropriation. In this paper, a multiplayer variant of Battle-of-the-Exes is suggested as appropriate for experimentation regarding fair and efficient coordination and turn-taking among selfish agents. Going beyond literature's fairness and efficiency, a novel measure is proposed for turn-taking coordination evaluation, robust to the number of agents and episodes of a system. Six variants of this measure are defined, entitled Alternation Measures or ALT. ALT measures were found sufficient to capture the desired properties (alternation, fair and efficient distribution) in comparison to state-of-the-art measures, thus they were benchmarked and tested through a series of experiments with Reinforcement Learning agents, aspiring to contribute novel tools for a deeper understanding of emergent social outcomes.

Keywords. Reinforcement Learning; Game Theory; Multi-agent Battle of the Exes; MBoE; Markov Games; Perfect Alternation; Fairness; Efficiency; Alternation Measures; ALT

1. Introduction

Multi-Agent Reinforcement Learning (*MARL*) can be adequate to model and explain complex conflictive situations like common-pool resource appropriation [1]. With a Markov Game [2] like modelling one can express situations that go beyond the minimal Game Theoretic frameworks. With *RL*, agents can learn selfishly and the equilibria reached can be studied [1]. Many measures (taken from Economics) have been applied to study the characteristics of such equilibria like Efficiency, Fairness and Sustainability e.g. [1],[3],[4]. These measures, although being adequate to capture the general exploitation of reward and how it is spread among agents, fail to capture the temporal dynamics of how reward is exploited, i.e. by turn taking. For this purpose we define a minimal environment based on the BoE [4] for the computational examination of turn-taking coordination among multiple selfish Q-learning agents. As it was shown in [2], the literature's Fairness and Efficiency measures were found insufficient and non-indicative for the evaluation of the fair and efficient distribution of such multi-agent systems, as mainly, they can either be "blind" to unfairness or inefficiency. Instead, we introduce the Perfect Alternation (*PA*) equilibrium notion as an optimal case where many agents acquire the full reward successively, as a point of reference to describe such systems' emergent behaviors by their discrepancy from PA. Furthermore, 6 novel Alternation measures for evaluation are shortly defined. Some results are indicatively presented at the end, to showcase the proposed measures usage.

[1] Corresponding Author: Technology Department, Universitat Pompeu Fabra, Carrer de Roc Boronat 138, 08018 Barcelona, Spain. E-mail: marti.sanchez@upf.edu.

2. Multiplayer Interpretation of the Battle of the Exes game

For the *MBoE* version, we consider that *n* agents do cooperate, if and only if they successfully alternate to acquiring the max reward, as this would be the fairest and efficient distribution of the recourses. We suggest a minimal interpretation yet other interpretations can also be considered. To define the problem, the goal is to generalize BoE to a multi-player game-theoretic scenario so that the coordination of selfish agents can be tested experimentally in a minimally dynamic environment. The basic requirement to ensure consistency with the problem definition of the 2-players version is the conceptualization of episodic non-cooperative game-theoretic scenarios where every agent moves simultaneously in each round. When only one agent reaches a terminal state, it gets the higher possible payoff and the remaining *n-1* agents get 0, yet when all *n* players reach the terminal state (tie) no one gets a payoff. The above are considered to be the basic limitations to ensure that the problem definition is consistent with BoE. It is accepted here that every agent can only move one step at a time only at his own pathway of *m* possible positions, including the initial one. The only way that an episode is terminated is when **at least one** agent reaches the end of its pathway. As in [2], an episode cannot end in a round that everyone halts.

Furthermore, special attention is required to be given regarding the reward that will be acquired, in the case that only some of the agents *k* with $0<k<n$, manage to reach their goal-states and therefore, if the game will be zero-sum or not. For example: 3 agents ($n = 3$) compete over the high payoff of 1 that can be acquired only if they reach the terminal state individually. In case that more than one agents reach the terminal state, they get only a low partial payoff, for example, $1/9$ ($p = 1/n^2$), and they get no payoff if all of them reach it together. This minimally dynamic version was carried out included 3 positions per agent. The initial, an intermediate, and a top one.

3. Perfect Alternation Measures

Before proceeding, it would be useful to define and propose Perfect Alternation (PA) Equilibrium. A Perfect Alternation (PA) in repeated games is considered the Pareto-optimal Nash Equilibrium when **all *n* players, alternate successively to the state of the highest payoff, one-by-one, and episode-by-episode, in any order**. However, this order is ideally repeated intact every *n* episodes. This equilibrium is meant to be diversified from other types of Alternation Equilibria which can include turn-taking of players every any number of episodes, in groups or even asymmetrically for each agent/ group. Of course, there can be other types of special equilibria that can be fair and efficient, however, their definition as "alternation equilibria" can be questioned, as semantically maybe ”solid alternation” should imply fixed periodicity of turn-taking among agents as in [4]. More specific practical and theoretical purposes that motivated the distinction of PA from the rest, are analytically discussed in [2].

Alternation measures or *ALT*, calculate the discrepancy of a turn-taking system behaviour from the ideal case of PA. Specifically, they aim to capture the performance of agents' succession to terminal positions, measuring the weighted rate of successful alternation of winners. This repeats per all possible sequences of *n* episodes - called batches - to indicate the agents' coordination throughout all *v* episodes. Ideally, every agent should win once every *n* episodes. For this reason, the algorithm evaluates each batch of episodes *b,* which can be considered as an overlapping window of *n* (number of agents) size. Then, the normalized accumulated evaluations of batches are averaged by the number of batches. Of course, the total number of batches is always *b=v–(n–1)* so the number of agents *n* must always be at least equal to the number of episodes *v*.

Specifically, to evaluate each agent's alternation within each batch, first, a sub-measure/ weight *β* is calculated (*β* calculation is analytically explained below). In the best-case scenario, *β* is equal to *1* for each batch. Thus, for all versions of ALT that are introduced in [2], the optimal Alternation's value $\widehat{ALT}$ is equal to *1*, meaning that in such a case all *n* agents exclusively won in succession, one per time throughout all *v* episodes. However, for each agent that wasn't included in the list of winners within *n* episodes of a batch, the evaluation measure *β* of this batch is reduced. In tie cases, that more than one winners occur in one episode, the algorithm splits their evaluation weight *β*, in a different manner according to the version of *ALT* that is used. $ALT^n = \frac{\sum_{j=1}^{b} \beta_j}{b}$, where *j* is the integer id of each batch, *b* is the number of possible batches within ν episodes and *β* is the sub-measure that evaluates each batch's alternation of agents accumulatively. *β* weights vary depending on which the version is measured.

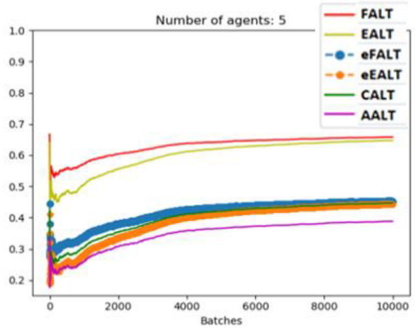
Fractional Alternation Measure (FALT) weights each batch with a fraction of the number of individual agents that managed to reach their terminal position at least once denoted, by the total of all the cases that an agent reached it, within this batch *j*: $\beta_j^{FALT} = \frac{f_j}{t_j}$ with *f* denoting how many agents out of *n* appeared in their terminal positions at least once within this batch of episodes*, t* denoting all the terminal occurrences*, j* the id of this batch. ***Exponential Fractional Alternation Measure (eFALT)*** uses the exponential version (square of) β_j^{FALT}. It is proposed for the cases when "stricter" evaluations of low alternation (as defined for FALT) are preferred, while more "generous" evaluations of higher alternation fit better to the given problem.

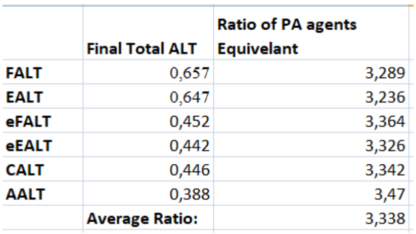
Exclusive Alternation (EALT) measures the rate which agents exclusively win within each episode of a batch. It is not that tolerant as FALT because only one agent should win at each episode. Otherwise the whole episode will be evaluated with *0* for all the batches that are included. The *β* value is calculated as $\beta_j^{EALT} = \frac{w_j * f_j}{n^2}$ where w_j is the number of winning episodes with an exclusive winner within the *j*th batch of episodes. ***Exponential Exclusive Alternation Measure (eEALT)*** or β_j^{eEALT} is again the square of β_j^{EALT} for an exponential "treatment" of the evaluation as in *eFALT.*

Complete Alternation (CALT) is stricter than the last proposed versions of the ALT measure, as it assigns a weight of *0* to tie situations. Specifically, for each episode, it multiplies β^{eFALT} with the difference between the maximum number of possible agents that can reach their top position and the actual ones. Then it adds up all the weighted β^{eFALT} values of each episode to divide them with the number of episodes per batch, which is always equal to n, times the max possible weight, which is equal to *(n – 1)* when only one agent reached the top in an episode*,* to average each episode's performance. This way, the fewer the winners the more the weight of β^{eFALT} for each episode. In the extreme case of a tie, this episode is evaluated with 0, affecting the batch's evaluation. Its *β* batches weights are calculated as follows: $\beta_j^{CALT} = \frac{\sum_{k=1}^{n}\left((n-Y_k) * \beta_j^{eFALT}\right)}{n*(n-1)}$, where integer *k* indicates the id of an episode within batch *j* with *0≤k≤n-1* and *n* is the number of agents. Also, *Y* is an *n*-sized array whose each element contains the number of agents who reached their top, for every episode *k* of batch *j.*

Absolute Alternation (AALT) is the last and the most sensitive measure to Alternation, as its changes are dramatic depending on the alternation phenomenon. It is expected to always be lower or equal to the rest of the Alternation measures. It assigns any non-exclusive winning position of an agent within a batch, with a weight of *0,* and takes into account only the successful alternations of exclusive winners of a batch. It gets *1* only in the ideal case that all agents win at least and only once per batch of episodes. Its β batches weights are calculated easily as follows: $\beta_j^{AALT} = \frac{g_j}{t_j}$, where g_j is the number of unique exclusive winnings of all agents within the whole j^{th} batch.

All the ALT measures values *X* are suggesting a level of alternation *A(X)* or *AltRatio*. This ratio or percentage is calculated by a function of the number of agents and the coefficients which are estimated by a environment-specific model-fitting regression. For this regression, extreme cases have been evaluated to be used as benchmarks for each version of ALT measure, as if *x* ∈*[2,40]* agents were perfectly alternating among each other to their top positions, while the rest *n-x* did not move from their initial positions, as analytically explained in [2]. Thus *A(X)* is indicative in terms of the equivalent of how many agents would perfectly alternate if all the rest were not moving at all, as shown in **Figure 1**. Thus, the outcome of those values does not necessarily mean that 3,338 agents out of 5 were indeed perfectly alternating, yet that the collective behavior is the equivalent of such a turn-taking throughout the whole experiment. Indicatively, some results out of a series of 41 experiments are:





	Final Total ALT	Ratio of PA agents Equivelant
FALT	0,657	3,289
EALT	0,647	3,236
eFALT	0,452	3,364
eEALT	0,442	3,326
CALT	0,446	3,342
AALT	0,388	3,47
	Average Ratio:	3,338

Figure 1. The final estimation of ALT version have an average error of *~0.1* for *5* agents and *10000* episodes. Because the *avg. AltRatio* is *>65%,* it is expected that the exponential versions eFALT & eEALT will be higher than FALT & EALT. Also, AALT shows the highest value as its evaluation has the most abrupt curve.

References

[1] J. Perolat, J.Z. Leibo, V. Zambaldi, C. Beattie, K. Tuyls, and T. Graepel. A multi-agent reinforcement learning model of common-pool resource appropriation, *Adv. Neural Inf. Process. Syst.* 2017-Decem (2017) 3644–3653.

[2] N. Papadopoulos. Study of turn-taking coordination for nagents in game-theoretic scenarios, with reinforcement learning: Proposal of an evaluation framework of Perfect Alternation Equilibria for multi-agent environments, Universitat de Pompeu Fabra, 2020. https://repositori.upf.edu/handle/10230/46270 .

[3] M.J. Gasparrini, and M. Sánchez-Fibla. Loss Aversion Fosters Coordination in Independent Reinforcement Learners, *Front. Artif. Intell. Appl.* 308 (2018) 307–311. doi:10.3233/978-1-61499-918-8-307.

[4] R.X.D. Hawkins, and R.L. Goldstone. The formation of social conventions in real- Time environments, *PLoS One*. 11 (2016) 1–14. doi:10.1371/journal.pone.0151670.

This page intentionally left blank

Computer Vision

This page intentionally left blank

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210147

On Determining Suitable Embedded Devices for Deep Learning Models

Daniel PADILLA [a,b], Hatem A. RASHWAN [a] and Domènec Savi PUIG [a]
[a] *DEIM, Universitat Rovira i Virgili, Tarragona, Spain*
[b] *Department of Research & Development, Quercus Technologies, Reus, Spain*

Abstract. Deep learning (DL) networks have proven to be crucial in commercial solutions with computer vision challenges due to their abilities to extract high-level abstractions of the image data and their capabilities of being easily adapted to many applications. As a result, DL methodologies had become a de facto standard for computer vision problems yielding many new kinds of research, approaches and applications. Recently, the commercial sector is also driving to use of embedded systems to be able to execute DL models, which has caused an important change on the DL panorama and the embedded systems themselves. Consequently, in this paper, we attempt to study the state of the art of embedded systems, such as GPUs, FPGAs and Mobile SoCs, that are able to use DL techniques, to modernize the stakeholders with the new systems available in the market. Besides, we aim at helping them to determine which of these systems can be beneficial and suitable for their applications in terms of upgradeability, price, deployment and performance.

Keywords. Embedded systems, Deep Learning, FPGA, GPU, DSP, SoC

1. Introduction

Deep learning (DL) has been considered to be one of the most cutting-edge Artificial intelligence (AI) techniques. However, AI companies must overcome a huge problem that they must upgrade their old systems based on traditional ML and computer vision techniques to successful and established products with new features based on DL. Old embedded systems, such as surveillance cameras, are the ones that can make more benefits from DL techniques. And, even though some intelligent cameras are able to process images with lightweight ML algorithms, but these systems have many times low precision for complex problems, e.g., object detection. Thus, upgrading such systems into something more intelligent and reliable that can actually extract information from the input data is the next target for many AI companies.

Not only the high precision achieved by the DL solutions and the low effort to be accomplished comparing to manual engineered (hand-crafted) solutions make them a promising technology. But also, the acquisition and storage of data become more available. The only drawback for using these techniques is the significant increase in the computation complexity that not all commercial applications can afford.

Since DL training is a heavy computational task, the mainstream research device is to use GPUs. Ordinarily, a PCIe card fit for a PC or Server. However, with devices with limited resources, such as CPU, Memory or even space, and the requirements for high throughput like real-time solutions, DL solutions seem to be a bit far away until now. For example, surveillance cameras can not be expanded with a PCIe card to use GPUs. Not only because they will not have enough space in its case, but also because inserting a PCIe card would mean redesign the whole camera hardware with lots of implications. As a result, some applications are not yet ready to work with DL techniques. However, the technology giants are moving towards more capable tools of making DL integration possible even in a mobile phone.

Several works are focusing on the main hardware options, however some of them are partial and others are not focused enough on the commercial side. For example, several surveys have focused on specific devices, such as Nvidia Jetson Series, with an overview of other devices like FPGAs [16]. Regarding FPGAs, there also are other kinds of surveys that involve several domains tagging and taxonomy [3,18].

In this work, we will first state the main problem related to the performance of DL models on embedded systems. We will then list and overlook some of the alternatives to the current systems. After that, we will state some of the characteristics of every device. Finally, making a decision based on some defined parameters to help the stakeholders to determine which of these embedded systems can be beneficial and suitable for their applications in terms of upgradeability, price, deployment and performance characteristics.

2. Methodology

We will analyse the current status of common embedded devices, such as FBGA, GPUs, CPUs, DSP, etc. Besides, we will define some formulae that can help in adapting an embedded product by adding or replacing some components to substitute the traditional computer vision solution with a DL solution. Thus, to precisely evaluate each embedded device and offer the reader the possibility to adapt the decision to its necessities, and based on a weighting value λ that factor score from 0 to 1, we evaluate each device with four different factors:

Upgradeability (U): For each embedded system to be upgraded, this factor measures the system ability, which means to move from the previous system to one capable of using DL techniques in terms of hardware changes. We use this ranking formula for weighting redesigns on the system:

$$U = (1-\gamma)\lambda_{U0} + \gamma\lambda_{U1},$$

where λ_{U0} stands for a subjective value for the number of changes required for the system design to include a new device and λ_{U1} measures the number of components that have to be changed to include this new device. γ is a weighting value between 0 to 1.

In this work, we set $\gamma = 0.4$, since redesigning the whole system to introduce a new block should penalize more than adapting the printed circuit board (PCB) of that device.

Deployment (D): This factor is related to software difficulties when applying the developed DL model to the target device. On the formula, we equally weight the number of frameworks (TensorFlow, Pytorch, etc.) being used (λ_{D0}) and the time used for different operations, such as compiling, compressing or readjusting the DL model to the deployment mode (λ_{D1}) from the checkpoint mode. As a rule of thumb, we set:

$$\lambda_{D0} = \begin{cases} 1 & \text{if } n = 1 \\ 0.5 & \text{if } n = 2 \\ 0 & \text{if } n > 2 \end{cases}, \qquad \begin{cases} 1 & \text{No need for operations} \\ 0.5 & \text{Few and quick operations} \\ 0 & \text{Operations takes long time} \end{cases},$$

$$\lambda_{D1} =$$

where n is the number of frameworks being used.

$$D = (1 - \alpha)\lambda_{D0} + \alpha\lambda_{D1},$$

where α is a weighting value between 0 to 1.

In this work, we set $\alpha = 0.6$, since we consider the compilation time and modification of the model (compression, etc.) are slightly more important than the number of frameworks used.

Price (P): Since every company has a target for its budget to decrease the final cost of a low-end product. The P factor ranks the prices for the embedded device required to use. Assume the target price of the required device is T_p set in this workaround 100 Eur that can be considered the target price of most standard distributed architectures of embedded systems of AI companies. Consequently, we will use the following formula to evaluate the price factor:

$$P = 1 - ((mean(p) - T_p)/max(p)),$$

where p is the list of prices for available devices that can be used in the new target systems. The P value will be close to 1, if the $mean(p)$ is close to the target price. Thus, the devices with low prices will have a high value, while low values will be related to expensive devices.

Performance (A): Finally, this factor measures the precision and speed (inference time or frames per second) of the new device. Since various devices will individually be tested, even with the same model (different frameworks, quantization, device's optimal adjustments, etc.), we need to know the variability between the performance on a PCIe GPU and the selected device. For that, we compute the Performance factor based on:

$$A = (1 - \theta)\lambda_{R0} + \theta\lambda_{R1},$$

$$\lambda_{R0} = \begin{cases} 1 & \text{if } \lambda_{Inf} <= 1.5 \\ 0.5 & \text{if } 1.5 < \lambda_{Inf} < 2 \\ 0 & \text{if } \lambda_{Inf} >= 2 \end{cases}, \qquad \lambda_{Inf} = \frac{\sum_{i=1}^{j} \log Inf_i}{j}$$

where Inf_i stands for each inference time of Table 2 for that device, and j is the number of available inferences in the table for the same device.

$$\lambda_{R1} = \begin{cases} 1 & \text{if } \lambda_{Prec} <= 1 \\ 0.5 & \text{if } 1 < \lambda_{Prec} < 5 \\ 0 & \text{if } \lambda_{Prec} >= 5 \end{cases}, \qquad \lambda_{Prec} = \frac{\sum_{i=1}^{k} (P_{PCIe_i} - P_i)}{k}$$

where P_{PCIe_i} is the precision value of the DL system on PCIe GPU devices, P_i is the precision values in Table 2 for each device, and k is the number of available precision values in the table for each device.

In this work, we set $\theta = 0.2$. Since performances values are more subjective to compression and model modifications done to fit into low hardware.

3. DL Networks study

Deep Neural Networks (DNN) are known for their large quantity of parameters and operations, such as enormous quantities of trainable parameters. That is translated into a complex structure and, in terms of memory, a big data bulk. In contrast, embedded systems tend to reduce the capacity in terms of memory and computation time. The most common deep convolutional neural networks (DCNN) on low-end embedding systems for object classification are VGG-16 [23] and MobileNet [22] and their variations. These networks are characterized by having low complexity and good enough precision compared to other deep architectures with more precision. In addition, there are three imposing families of Object Detection architectures: YOLO V1 to V5 [21], RCNN [6], and SSD [15,22]. The fastest algorithms amongst them is those based on YOLO.

There is a constant flow of research to make the DNN compact, quicker or simpler to reduce the overall resources needed for a DL inference application. These works depend on two different main lines. The first line is to reduce the number of connections, parameters, and architecture to reduce the model complexity. Examples of such types of LD models are tuning networks to get simpler ones, such as in MobileNet V2 [22]. While the other works go to compress the DNN networks [8]. In turn, the second line focuses on changing the operations insides of DNN networks to get quicker and/or more efficient operations. For instance, some works used Fourier Transforms [13], and binarized models [11] or even they have modified internal network architectures looking for faster operations [5].

In this work, we use the most common DNN networks to provide a fair comparison between several embedded devices. The tested networks are MobileNetv2, VGG-16 and VGG-19 for Object Classification, and YOLO tiny, YOLOv2, YOLOv3 and SSDLite+MobileNetv2 for Object Detection.

4. Hardware

Since the main focus of this paper is to enhance an AI physical product using DL, the ideal solution would be simply changing the Integrated Circuit of the product allowing us to improve the product with DL. That is not often possible since changing an IC usually means changing many more hardware components. Different devices and some specific classifications can allow us to work with DL techniques. We will focus on the most desirable embedded system by AI companies: GPUs, FPGAs and NPUs in turn, ASIC, CPUs and DSPs will be out of scope in this paper.

To be able to compare the different market devices, we have added Table 2 with a compilation of various embedded systems with comparable experiments using trained DL models. We performed three Object Classification tests and 4 Object Detection tests based on well-known DNN network references.

For Object Classification, the most referenced models are MobileNetV2, VGG-16 and VGG-19 that were trained using the ImageNet dataset. The results are given with two factors: the inference time ($Inf.(ms)$) expressed in milliseconds and Top-1 accuracy ($Top1$), as shown in Table 2). For the Object Detection problem, the referenced models are YOLO tiny, YOLOv2, YOLOv3, SSDLite-MobileNetV2 that were trained with the

Device Type		CPU	Nvidia PCIe GPU				Nvidia Jetson Series				FPGA					Mobile SoC					
Model		Intel Xeon E5-2650	Titan X	RTX 2080 Ti	GTX 1080	GTX 1050 Ti	Nano	Tx1	TX2	AGX	Xilinx Virtex 7	Zynq SoC Z-7100	Arria 10 GX	Cyclone V	Stratix V	Kirin 990 5G	Snapdragon 855+	Snapdragon 835	Snapdragon 821	Snapdragon 820	Helio P22
MobileNetV21	Inf.(ms)	6.6	1	0.7	1	-	16	-	-	-	-	-	-	-	-	6	15	181	75	98	243
	Top1	71.9	71.9	71.9	71.9	-	71.9	-	-	-	-	-	-	-	-	69.6	70.5	71.9	72	71.9	71.9
VGG-16	Inf.(ms)	-	-	-	-	-	-	347	-	-	519	-	110	1928	254	-	-	-	-	-	-
	Top1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VGG-19	Inf.(ms)	-	-	0.6	-	-	100	-	43	7	-	-	-	-	-	42	182	3754	-	4995	7053
	Top1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
YOLO tiny	Inf.(ms)	-	5	-	-	-	-	150	58	-	125	-	15	-	-	-	-	-	-	-	-
	mAP(%)	-	57.1	-	-	-	-	-	32	57.1	-	-	57.1	-	-	-	-	-	-	-	-
YOLOv2	Inf.(ms)	-	15	-	-	-	-	-	172	-	-	80	54	-	-	-	-	-	-	-	-
	mAP(%)	-	76.8	-	-	-	-	-	51	-	-	69.1	76.1	-	-	-	-	-	-	-	-
YOLOv3	Inf.(ms)	133	-	-	31	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	mAP(%)	30	-	-	30	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
SSDLite-MobileNetV2	Inf.(ms)	-	-	-	28	16	26	-	-	-	-	-	18	-	-	-	-	-	200	-	-
	mAP(%)	-	-	-	-	77	-	-	-	-	-	-	73	-	-	-	-	-	-	-	-
References		$\ast^{1,2}$	$\ast^{1,3}$	$\ast^{1,4}$	$\ast^{1,2}$	$\ast^{5,2}$	$\ast^{6}$	$\ast^{7}$	$\ast^{8,9}$	$\ast^{10,1}$	$\ast^{7}$	$\ast^{12}$	$\ast^{5,13,14}$	$\ast^{12}$	$\ast^{13}$	$\ast^{1}$	$\ast^{1}$	$\ast^{1}$	$\ast^{15}$	$\ast^{1}$	$\ast^{1}$

[1] [9] [2] Omni-benchmarking Object Detection. [3] [30] [4] RTX 2080 Ti Deep Learning Benchmarks with TensorFlow. [5] [14] [6] Jetson Nano: Deep Learning Inference Benchmarks [7] [7] [8] Giant Leaps in Performance and Efficiency for AI Services, from the Data Center to the Network's Edge [9] [10] [10] Jetson AGX Xavier: Deep Learning Inference Benchmarks [11] [29] [12] [17] [13] [28] [14] [30] [15] [22]

Table 2. Comparison between different devices.

COCO dataset. For object detection, the result of inference time is maintained, but the mean Average Precision (*mAP%*) is used instead of Top-1 accuracy. Finally, we add an Estimated Price extracted for some of these devices and in cases of Mobile SoCs, the mobile associated.

4.1. GPUs

The first ideal solution is the one with less effort for the researchers by using GPUs. The solution is a training environment that the company can only transfer the trained DL model to a target device and execute it. The GPUs, such as the NVIDIA Titan series, can be easily integrated with a PC or even a dedicated server with dedicated PCI cards. But including these solutions could be an overshoot, because of the higher prices of these GPUs. Thus we can note this solution is not the preferable solution for the industry.

Model	Estimated Price (Eur)
Intel Xeon E5-2650	875
Titan X	1200
RTX 2080 Ti	1150
GTX 1080	500
GTX 1050 Ti	150
Nano	100
Tx1	300
Tx2 4G	272
TX2	416
TX2i	718
AGX 8G	618
AGX	909
Zynq SoC Z-7020	100
Xilinx Virtex 7	300
Zynq SoC Z-7100	3K
Arria 10 GX	800 - 2K5
Cyclone V	50
Stratix V	6K -16K
Stratix 10 GX	7K - 40K
Kirin 990 5G	800$^{\mathrm{m}}$
Snapdragon 855+	420$^{\mathrm{m}}$
Snapdragon 835	150$^{\mathrm{m}}$
Snapdragon 821	100$^{\mathrm{m}}$
Snapdragon 820	75 $^{\mathrm{m}}$
Helio P22	100$^{\mathrm{m}}$

Table 1.: Prices of different devices. Where $^{\mathrm{m}}$ stands for the price listed for the mobile using this SoC.

[2] https://towardsdatascience.com/omni-benchmarking-object-detection-b390cc4114cd

[4] https://lambdalabs.com/blog/2080-ti-deep-learning-benchmarks/

[6] https://developer.nvidia.com/embedded/jetson-nano-dl-inference-benchmarks

[8] https://www.nvidia.com/content/dam/en-zz/Solutions/Data-Center/tesla-product-literature/t4-inference-print-update-inference-tech-overview-final.pdf

[10] https://developer.nvidia.com/embedded/jetson-agx-xavier-dl-inference-benchmarks

However, NVIDIA has detected this niche and has promptly developed some smaller units for embedded devices. We are talking about the Jetson Series: Mini-Computer modules with a specific power needed to execute some DL networks. These low-end Jetson GPU modules allow some CUDA capabilities, being very easy to run the same model as your high-end desktop PC's GPU. The computational power and memory of these series limit the complexity of DL models.

With lower prices of the Jetson modules is not weird to see a large number of academic developments focused on Jetson Nano. That is very popular in robotics [24,25,31], also in other applications like adapting YOLO[21] network to fit into this low-end module [29]. Other modules like Jetson TX2 is used in medical image-analysis [19] or in other cases, such as detecting ships [32]. But since the price goes up, as expected, the research tends to lessen, although different benchmarks from unofficial parts [9,1][1][2] and official ones[3][4] seems to indicate their good performance for real-time embedded systems. All in all, Jetson modules seems a good solution for upgrading embedded system.

Table 2 listed different capabilities for computer power evaluation and approximated prices for some of NVIDIA's GPUs and NIVIDIA Jeston series.

4.2. FPGAs

Recently, there are many discussions about another desirable device for applying DL techniques[5], which is related to the Field Programmable Gate Arrays (FPGAs). Since their output results are not instructions-based anymore. Several articles have been presented for improving even further FPGAs. performance [27,12]. However, one of its main important flaws is in the compilation and programming experience. Although nowadays there are possible frameworks for programming FPGAs [20,4], which provide us with a better programming experience for FPGAs, these frameworks are on top of the other DL frameworks (i.e., Tensorflow and Pytorch). Therefore, these FPGA frameworks are not at the same accessibility as CPU/GPU cores yet. Besides, FPGA needs to be reprogrammed for every little change on the DL models, contrary to the GPUs where the change is made on memory and GPU has not to be changed.

The historical weak point of FPGA was the floating-point operations. That is maintained with the help of specialized DSPs blocks embedded in FPGAs to enhance floating-point operations. These blocks had granted a big impact of FPGAs on DL techniques. Thus, in combination with high memory bandwidth and very fast response, FPGAs become a tough competitor for NVIDIA's GPUs. Indeed, large companies like Microsoft and Google move towards these solutions. There are already some interesting applications achieved on cheaper FPGAs, e.g., Zynq 7000, like Facial Expression recognition [33], and Underwater real-time image recognition [33], and other research available on

[1] RTX 2080 Ti Deep Learning Benchmarks with TensorFlow: https://lambdalabs.com/blog/2080-ti-deep-learning-benchmarks/

[2] NVIDIA Jetson AGX Xavier Benchmarks: https://www.phoronix.com/scan.php?page=article&item=nvidia-jetson-xavier&num=4

[3] 2019 Machine Learning Benchmark: https://www.nvidia.com/en-us/data-center/2019-machine-learning-benchmarks/

[4] Jetson AGX Xavier: Deep Learning Inference Benchmarks: https://developer.nvidia.com/embedded/jetson-agx-xavier-dl-inference-benchmarks

[5] Why use an FPGA instead of a CPU or GPU? https://blog.esciencecenter.nl/why-use-an-fpga-instead-of-a-cpu-or-gpu-b234cd4f309c

benchmarks with Intel's Arria 10 [14]. Promising results with FPGAs have been summarised in Table 2.

However, there exist a large gap between low-end and high-end FPGAs price. For instance, high-end FPGAs use SoCs that are used in GPUs. These SoCs tend to have more computational Power per Euro ratio[1]. It is noted that any embedded system with a big budget for upgrading will get the most benefits in terms of performance and efficiency, as shown in Table 3.

4.3. NPUs

The final targets are those low powered and low-cost Mobiles. Many manufacturers had proposed their solutions for mobiles and smartphones. For instance, tensor processing unit, neural network processor, intelligence processing unit, vision processing unit and graph processing unit are some of the names doted by manufacturers known more globally by Neural Processor Unit (NPU). These SoCs have less performance than dedicated GPUs, however, with their tiny modules and cost, they can be more suitable and affordable solutions for real-time applications for embedded systems.

Until recently, mobile applications with such kind of AI applications were server-based. These applications packed and sent the input data (e.g., voice or video) to a dedicated server to make the inference. This could take some time and real-time applications will be more difficult. Nowadays, we have several SoCs with DL features to allow us to jump that barrier of online applications and execute the DL model in the same smartphone [26]. In this case, we can, for example, execute YOLO for object detection in an iOS mobile [2] and the inference is done inside an iOS system and not in a cloud-based server.

There are several new SoCs[9] that gives us a similar performance to old GPUs and even better performance than CPUs. Table 2 summarise some of these values with the MobileNetV2 network. We can observe that inference times can go under 10 msec with the latest SoC at the expense of 2 percentiles in precision, while some of the precision values near the CPU baseline can go up to speeds of 75 milliseconds per inference, which should be enough for some real-time applications. Being able to get prices for SoCs is very difficult.Usually, manufacturers do not give prices for a single SoC, and their values are limited. However, we can compare the cost of the mobiles themselves that come with these SoCs. For example, Table 2 shows some current prices for some mobiles. The prices range is between 20-200 Euros, with a mean of 80 Eur.

5. Discussion

In this paper, we have reviewed the state of the art including the system upgrading from traditional AI systems to promising DL-based systems. For the industry, upgrade an AI embedded system has its cost. However, the high precision, which could achieve with DL, may well deserve it. The community is also going through many different types of

[1] GPU vs FPGA Performance Comparison White Paper 2: `https://www.bertendsp.com/pdf/whitepaper/BWP001_GPU_vs_FPGA_Performance_Comparison_v1.0.pdf`

		GPU	FPGA	NPU
U	$\lambda_U 0$	0	0.5	1
	$\lambda_U 1$	0.5	0.5	0.75
	Total	0.1	0.5	0.9
D	λ_{D0}	1	2+	2
	λ_{D1}	1	0	0.75
	Total	1	0.1	0.5
P	Total	0.41	0.19	0.95
A	Total	0.5	0.8	0.6
E		2.01	1.59	2.95

Table 3.: Evaluation factors values for each SoCs based on four factors.

research to improve devices to run more complex models, i.e., DL models, and improve the efficiency of the networks. We depend on the four factors mentioned in section 2 (i.e., Upgradeability (U), Deployment (D), Price (P) and Performance (A)) to evaluate the three well-known embedded ystems: GPUs, FPGAs and NPUs. The final evaluation factor (E) can be defined as a weighted sum of the four factors:

$$E = \xi_1 U + \xi_2 D + \xi_3 P + \xi_4 A,$$

where we set $\xi_1 = \xi_2 = \xi_3 = \xi_4 = 1$. As shown Table3, we presented quantitative results of the three embedded systems with the four factors. NPUs provided the best evaluation value of 2.95 in terms of U, D, P and A factors. In turn, with FPGAs, the evaluation value was degraded to 1.59. Although FPGAs provided the best performance value (A = 0.8) among the three systems. Similarly,in Figure 1, the results show NPUs should be the best target embedded system for AI companies, followed by GPUs and finally FPGAs supporting our discussion.

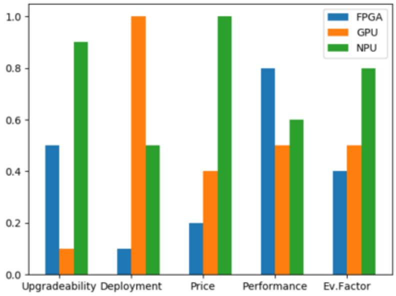


Figure 1.: FPGA, GPUs, NPUs evaluation scores.

The mobile market is a great push for the DL embedding sector. It may not have the best performance, and some kind of trade-off is usually needed when choosing between these devices, but the prices of their ICs, the current investment of the sector in DL and the deployment, may well make these ICs the best ones to be used when upgrading old embedded products.

As things are, this mobile sector will be the ones which, most probably, will quickly grow in the coming years due to the growing demand. Besides, even though other solutions may suit better when they are appropriately chosen, they are much more application-specific. A good example would be where there is no need for an operating system, which the FPGA could fit quite well. Or a large-case product with already I/O pins could be easier to adapt with an NVIDIA Jetson SoM. Mobiles are already capable of running low-end models. And the complexity of these models will keep increasing as the sector moves toward using more AI locally. We can add the low price of devices compared to other ones and, although we would think that low precision and speed may be limited. Thus, we could use several methods to make the trade-off:

- Limit the speed: not all embedded systems must be real-time
- Accept a little loss: The cost could compensate the little loss
- Use the lower cost: The system could upgrade and improve weak points (i.e. using more sensors, extra memory, etc)

References

[1] Mario Almeida, Stefanos Laskaridis, Ilias Leontiadis, Stylianos I Venieris, and Nicholas D Lane. EmBench: Quantifying Performance Variations of Deep Neural Networks across Modern Commodity Devices. 2019.

[2] Maneesh Apte, Simar Mangat, and Priyanka Sekhar. Yolo net on ios. Technical report, 2017.

[3] Ahmed Ghazi Blaiech, Khaled Ben Khalifa, Carlos Valderrama, Marcelo A.C. Fernandes, and Mohamed Hedi Bedoui. A Survey and Taxonomy of FPGA-based Deep Learning Accelerators, sep 2019.

[4] Roberto Di Cecco, Griffin Lacey, Jasmina Vasiljevic, Paul Chow, Graham Taylor, and Shawki Areibi. Caffeinated FPGAs: FPGA framework for convolutional neural networks. In *Proceedings of the 2016 International Conference on Field-Programmable Technology, FPT 2016*, pages 265–268. Institute of Electrical and Electronics Engineers Inc., may 2017.

[5] Wei Ding, Zeyu Huang, Zunkai Huang, Li Tian, Hui Wang, and Songlin Feng. Designing efficient accelerator of depthwise separable convolutional neural network on FPGA. *Journal of Systems Architecture*, 97:278–286, aug 2019.

[6] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation Tech report (v5). Technical report, 2014.

[7] Kaiyuan Guo, Lingzhi Sui, Jiantao Qiu, Song Yao, Song Han, Yu Wang, and Huazhong Yang. From model to FPGA: Software-hardware co-design for efficient neural network acceleration. In *2016 IEEE Hot Chips 28 Symposium, HCS 2016*. Institute of Electrical and Electronics Engineers Inc., may 2017.

[8] Song Han, Huizi Mao, and William J. Dally. Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding. oct 2015.

[9] Andrey Ignatov, Radu Timofte, William Chou, Ke Wang, Max Wu, Tim Hartley, and Luc Van Gool. AI Benchmark: Running Deep Neural Networks on Android Smartphones. Technical report, 2018.

[10] Duseok Kang, Dong Hyun Kang, Jintaek Kang, Sungjoo Yoo, and Soonhoi Ha. Joint optimization of speed, accuracy, and energy for embedded image recognition systems. In *Proceedings of the 2018 Design, Automation and Test in Europe Conference and Exhibition, DATE 2018*, volume 2018-January, pages 715–720. Institute of Electrical and Electronics Engineers Inc., apr 2018.

[11] Jaeha Kung, David Zhang, & Gooitzen Van Der Wal, Sek Chai, and Saibal Mukhopadhyay. Efficient Object Detection Using Embedded Binarized Neural Networks. 2018.

[12] Meng Jhe Li, An Hong Li, Yu Jung Huang, and Shao I. Chu. Implementation of deep reinforcement learning. In *ACM International Conference Proceeding Series*, volume Part F1483, pages 232–236. Association for Computing Machinery, 2019.

[13] Sheng Lin, Ning Liu, Mahdi Nazemi, Hongjia Li, Caiwen Ding, Yanzhi Wang, and Massoud Pedram. FFT-based deep learning deployment in embedded systems. In *2018 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, volume 2018-Janua, pages 1045–1050. IEEE, mar 2018.

[14] Zhongyi Lin, Matthew Yih, Jeffrey M. Ota, John D. Owens, and Pinar Muyan-Ozcelik. Benchmarking Deep Learning Frameworks and Investigating FPGA Deployment for Traffic Sign Classification and Detection. *IEEE Transactions on Intelligent Vehicles*, 4(3):385–395, sep 2019.

[15] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng Yang Fu, and Alexander C. Berg. SSD: Single shot multibox detector. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 9905 LNCS, pages 21–37. Springer Verlag, 2016.

[16] Sparsh Mittal. A Survey on optimized implementation of deep learning models on the NVIDIA Jetson platform, aug 2019.

[17] Hiroki Nakahara and Tsutomu Sasao. A High-speed Low-power Deep Neural Network on an FPGA based on the Nested RNS: Applied to an Object Detector. In *Proceedings - IEEE International Symposium on Circuits and Systems*, volume 2018-May. Institute of Electrical and Electronics Engineers Inc., apr 2018.

[18] Razvan Nane, Vlad Mihai Sima, Christian Pilato, Jongsok Choi, Blair Fort, Andrew Canis, Yu Ting Chen, Hsuan Hsiao, Stephen Brown, Fabrizio Ferrandi, Jason Anderson, and Koen Bertels. A Survey and Evaluation of FPGA High-Level Synthesis Tools. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 35(10):1591–1604, oct 2016.

[19] Bojan Nokovic and Shucai Yao. Image Enhancement by Jetson TX2 Embedded AI Computing Device. pages 1–4. Institute of Electrical and Electronics Engineers (IEEE), jul 2019.

[20] Alexandros Papakonstantinou, Karthik Gururaj, John A. Stratton, Deming Chen, Jason Cong, and Wen-Mei W. Hwu. FCUDA: Enabling efficient compilation of CUDA kernels onto FPGAs. In *2009 IEEE*

7th Symposium on Application Specific Processors, pages 35–42. IEEE, jul 2009.

[21] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You Only Look Once: Unified, Real-Time Object Detection. jun 2015.

[22] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang Chieh Chen. MobileNetV2: Inverted Residuals and Linear Bottlenecks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 4510–4520, jan 2018.

[23] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. sep 2014.

[24] Siddhartha S. Srinivasa, Patrick Lancaster, Johan Michalove, Matt Schmittle, Colin Summers, Matthew Rockett, Joshua R. Smith, Sanjiban Choudhury, Christoforos Mavrogiannis, and Fereshteh Sadeghi. MuSHR: A Low-Cost, Open-Source Robotic Racecar for Education and Research. aug 2019.

[25] Teixeira, Nogueira, Dalmedico, Santos, Arruda, Neves-Jr, Pipa, Ramos, and . Intelligent 3D Perception System for Semantic Description and Dynamic Interaction. *Sensors*, 19(17):3764, aug 2019.

[26] Marian Verhelst and Bert Moons. Embedded Deep Neural Network Processing: Algorithmic and Processor Techniques Bring Deep Learning to IoT and Edge Devices. *IEEE Solid-State Circuits Magazine*, 9(4):55–65, 2017.

[27] Chao Wang, Lei Gong, Qi Yu, Xi Li, Yuan Xie, and Xuehai Zhou. DLAU: A scalable deep learning accelerator unit on FPGA. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 36(3):513–517, mar 2017.

[28] Dong Wang, Ke Xu, and Diankun Jiang. PipeCNN: An OpenCL-based open-source FPGA accelerator for convolution neural networks. In *2017 International Conference on Field-Programmable Technology, ICFPT 2017*, volume 2018-January, pages 279–282. Institute of Electrical and Electronics Engineers Inc., feb 2018.

[29] Alexander Wong, Mahmoud Famuori, Mohammad Javad Shafiee, Francis Li, Brendan Chwyl, and Jonathan Chung. YOLO Nano: a Highly Compact You Only Look Once Convolutional Neural Network for Object Detection. oct 2019.

[30] Xianchao Xu and Brian Liu. FCLNN: A Flexible Framework for Fast CNN Prototyping on FPGA with OpenCL and Caffe. In *Proceedings - 2018 International Conference on Field-Programmable Technology, FPT 2018*, pages 241–244. Institute of Electrical and Electronics Engineers Inc., dec 2018.

[31] Kailun Yang, Xinxin Hu, Hao Chen, Kaite Xiang, Kaiwei Wang, and Rainer Stiefelhagen. DS-PASS: Detail-Sensitive Panoramic Annular Semantic Segmentation through SwaftNet for Surrounding Sensing. Technical report, 2019.

[32] Hongwei Zhao, Weishan Zhang, Haoyun Sun, and Bing Xue. Embedded deep learning for ship detection and recognition. *Future Internet*, 11(2), 2019.

[33] Minghao Zhao, Chengquan Hu, Fenglin Wei, Kai Wang, Chong Wang, and Yu Jiang. Real-time underwater image recognition with FPGA embedded system for convolutional neural network. *Sensors (Switzerland)*, 19(2), jan 2019.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210148

Prostate Cancer Delineation in MRI Images Based on Deep Learning: Quantitative Comparison and Promising Perspective

Eddardaa BEN LOUSSAIEF [a,1], Mohamed ABDEL-NASSER [a] and Domènec PUIG [a]
[a] *Department of Computer Engineering and Mathematics, Universitat Rovira i Virgili, 43007 Tarragona, Spain*

Abstract. Prostate cancer is the most common malignant male tumor. Magnetic Resonance Imaging (MRI) plays a crucial role in the detection, diagnosis, and treatment of prostate cancer diseases. Computer-aided diagnosis systems can help doctors to analyze MRI images and detect prostate cancer earlier. One of the key stages of prostate cancer CAD systems is the automatic delineation of the prostate. Deep learning has recently demonstrated promising segmentation results with medical images. The purpose of this paper is to compare the state-of-the-art of deep learning-based approaches for prostate delineation in MRI images and discussing their limitations and strengths. Besides, we introduce a promising perspective for prostate tumor classification in MRI images. This perspective includes the use of the best segmentation model to detect the prostate tumors in MRI images. Then, we will employ the segmented images to extract the radiomics features that will be used to discriminate benign or malignant prostate tumors.

Keywords. Prostate cancer, MRI images, image segmentation, deep learning.

1. Introduction

Prostate cancer is the most common type of cancer among men worldwide. There were 1.3 million new cases only in 2018. The highest rates of prostate cancer in 2018 were mainly in the European countries. Age-adjusted incidence rates of prostate cancer have increased dramatically and this is large because of the increased availability of screening for prostate-specific antigen (PSA) which were conducted for men without symptoms of the disease. This screening leads to the detection of many prostate tumors that are small or would otherwise remain unrecognized, and which may or may not develop further into higher stage disease. Accurate delineation of prostate cancer using medical scanners plays a crucial role in prostate diseases diagnosis and treatment.

[1] Department of Computer Engineering and Mathematics, Universitat Rovira i Virgili, 43007 Tarragona, Spain; E-mail: eddardaa.eniso@gmail.com.

Previous research aimed at developing Computer-Aided Diagnosis (CAD) systems for Prostate cancer detection, classification, and prognostics using different medical imaging modalities, such as Computed Tomography (CT), Magnetic Resonance Imaging (MRI), and histopathological images. Early attempts prove that convolutional neural networks (CNNs) have achieved remarkable progress in many fields, particularly in computer vision and image understanding. Many researchers have also used CNNs to segment the prostate from MRI images. For instance, Milletari et al. [1] proposed a V-Net fully convolutional network (FCN) with the Dice loss for accurate prostate segmentation. Yu et al. [2] introduced a novel network that incorporates boundary estimation, feature extraction, and shapes prior to prostate detection.

Regarding the pitfalls of deep learning architectures or the data acquisition strategy of medical imaging. One could find that the literature is still discussing the potentials of deep learning-based approaches for medical image segmentation. Regardless of the progress achieved, automated prostate segmentation remains challenging and the essential part of the research has not been fully addressed. As it is noted that the outcomes do not fully fit the clinical needs in terms of the accuracy and the precision rate to detect cancer in an earlier stage.

This paper aims at proposing a comparison between state-of-the-art deep learning segmentation models for prostate cancer delineation in MRI images. We highlight the limitations and strengths of each segmentation model. We train the segmentation model on three public datasets: Promise12, ISBI Challenge2013, and ProstateX. DSC coefficient and Hausdorff Distance evaluation metrics are used to assess the performance of the prostate cancer delineation models. This paper presents the first stage of our prostate cancer CAD system. Our perspective consists of the use of the best segmentation model to detect the prostate tumors in MRI images. Then, we will employ the segmented images to extract the radiomics features that will be used to discriminate benign or malignant prostate tumors.

The remainder of the paper is structured as follows. Section 2 provides a review of literature. Section 3 describes the proposed methodology in detail and highlights our perspective. In section 4, provides various experiments on prostate tumor segmentation and detection. Finally, we present our conclusion and introduce future remarks in the last Section 5.

2. Related Work

Previous research has established promising results for automated prostate tumor segmentation. The developed methods can be mainly categorized into three classes: deformable methods, multi-atlas-based methods, and learning-based methods [1][2]. Firstly, deformable methods aim to accurately delineate the prostate, Toth et al. [4] introduced improved active appearance models (AAMs). Klein et al. [3] used atlas matching for prostate segmentation. The principal idea is to regroup the segmented images with target images and then fuse the aligned segmentation to reach the final results. Several deep-learning-based methods currently exist for automated prostate tumor delineation due to its automatic representation learning. Below, we present the 2D and 3D prostate segmentation methods.

2.1. 2D prostate segmentation methods

Studies over the past decade have proved that the use of Convolutional Neural Networks (CNNs) have achieved their goals in many domains, particularly in computer vision[5][6] and medical imaging analysis. Automated segmentation is one of the pillars of medical image analysis, many researchers have explored the power of CNN in prostate cancer segmentation [7] [9]. The U-Net [8] architecture is one of the most popular networks for medical imaging segmentation. It uses a CNN as a downward sampling path with an up-sampling operation to boost the resolution of the output feature maps. Furthermore, It uses skip connections to transfer more information to the feature maps that are placed within the up-sampling operations [8]. This could be noted in the work of Clark et al [10], who is considered to be among the first researchers who used the U-net to segment the whole prostate. The U-net has been adapted to segment the whole prostate gland and transitional zone in Diffusion-Weighted MRI (DWI). They were able to achieve promising results on an in-house dataset. However, when they tested their approach on T2-weighted (T2W) MRI data from the Promise 12 challenge [11] they were not able to achieve good accuracy.

Zhu et al. [12] proposed a deeply supervised U-Net architecture where they adjusted the number of convolution layers. The role of the deep supervision strategy is to supervise the hidden layers within the network and propagate them to the lower levels of the network. Also, they introduced residual blocks to reduce the number of hyper-parameters compared to the original U-Net. Tian et al. [13] implemented a PSNET that uses fully connected networks [14]. They tested their model on an in-house dataset consisting of 42 T2W MRI volumes and on two open-source prostate datasets: the ISBI2013 challenge [15] and the PROMISE12 challenge [11] datasets. Karimi et al. [16] used a smaller FCN consisting of three layers and validated it using a cross-validation scheme on a smaller dataset containing 49 T2W axial MRI images and 26 MRI images from the PROMISE12 challenge [11]. With this smaller network, Karimi et al. achieved a high DSC on their validation dataset.

In [17], Yoo et al. presented an automated CNN-based pipeline for detecting prostate cancer for an axial DWI (Diffusion-weighted imaging) image and each patient separately. Their proposed pipeline includes three stages. The first stage consists of classifying each DWI slice using five individually trained CNNs models. The second stage extracts the first-order statistical features (e.g., mean, standard deviation, median) from the CNNs outputs. Then, the relevant features are selected via a decision tree-based feature selector. In the last stage, they used a Random Forest classifier to classify patients. They used first-order statistical features to discriminate patients into sets with and without prostate cancer. Yu et al. [2] added a residual connection to the U-net network, which improves the prostate segmentation by using a sum operation instead of the concatenation operation features into the up-sampling layer. It turns the model into a hybrid model called ResNet-U-Net. To boost the accuracy of prostate delineation, Zhu et al. [18] used a cascaded U-Net, where the role of the first network is to segment the whole prostate gland, and the obtained segmented gland was fed into the last network to detect the peripheral zone.

Liu et al. [20] used an FCN with ResNet50 as the backbone of their network to detect the prostate zones with the mechanism of feature pyramid attention to capture relevant semantic information at multiple scales. The results obtained by the features pyramid attention procedure are combined to generate a high-resolution feature representation, which improves the segmentation results over the original U-Net [8]. To ameliorate the prostate's delineation, Nie and Shen Nie [21] introduced a semantic guided strategy to learn discriminative features. They used a soft contour constraint mechanism to model the blurry boundary and trained their network using the 5 k-fold cross-validation on 50 prostate cancer patients with ground truths annotated manually of the rectum, prostate, and bladder. Their results were promising and outperformed state-of-the-art techniques. Zhu et al.[22] proposed a novel boundary-weighted segmentation loss that boosts the accuracy of the boundary segmentation. They also used a boundary-weighted transfer learning approach (domain adaptation) to surmount the restriction of small training datasets. Thanks to this strategy and several datasets as the source and target domain, they achieved state-of-the-art results. In [25], the authors used a channel-wise feature recalibration and integrated squeeze and excitation blocks into their network to enhance the segmentation results [25]. Their work performed well when they trained the model on all of the introduced datasets, but their network was less robust using some datasets.

To alleviate the issue of limitation of the training datasets and obtain a large amount of training data for creating a robust segmentation model, Liu et al. [27] implemented a multi-site network by aggregating prostate MRIs from multiple sites. Their network was able to learn universal representation across heterogeneous MRI scanners and images. Furthermore, they introduced Domain-Specific Batch normalization layers to enable the network to estimate the statistics and perform normalization for each site it had been trained on, separately. David Gillespie et al. [30] proposed a review for deep learning-based methods for prostate segmentation in MRI images and discussed their limitations and strengths. To improve the prostate segmentation, they introduced an optimized 2D U-net that uses the Ranger optimizer [28] and Mish Activation [29].

2.2. 3D prostate segmentation methods

To accurately segment the prostate in MRI images, Milletari et al. [1] introduced a volumetric CNN based on V-shape fully convolutional networks (FCN). They adapted the U-Net architecture to segment prostates from 3D MRI volumes. They used Promise12 datasets to train their network. They showed that their network performed significantly better when they applied a data augmentation procedure as preprocessing step on the datasets. To improve the segmentation accuracy of 3D volumes, Wang et al. [19] introduced a novel deeply supervised model into a 3D U-Net with group dilated convolutions to automatically segment the prostate gland in MRI scans. The main reason to use the deep supervision mechanism is to avoid exploding or vanishing gradients during the training stage of deep models, which forces the hidden layers filters to support highly discriminate features. A group of dilated convolutions was used to extract more global contextual information, as they extend the receptive field of the network. They also used a combined loss function including cosine and cross-entropy that evaluates the similarity and dissimilarity between segmented and manual contours. Zavala Romero et al. [26] developed a 3D multi-stream U-Net and proposed data prepossessing to normalize the data given from each MRI vendor. They implemented six models (for each vendor and combined vendor datasets). The results demonstrated that the combined vendor models performed better in the case of peripheral zone detection. However, for detecting the whole prostate from MRI images the individual data models performed better.

To extract a piece of relevant information from the Z-axis, a 3D U-Net network was proposed by Isensee et al [23]. But, 3D-Unet leads to the use of small training data because it needs a performant memory for computation. To tackle this issue, Isensee et al [23] proposed a novel method that compresses domain knowledge and chooses the best nnU-Net (2D or 3D) model for a given dataset. Using 5 fold cross-validation scheme, they trained a 2D and a 3D variant nnU-net on the Promise12 challenge [11] and the Medical Decathlon dataset [24]. One common restriction of the mentioned work is that Isensee et al used a single site Prostate MRI scans to train their models, and get promising results on a private dataset. But it may not perform well on datasets from various sites or MRI vendors. This is a crucial challenge for prostate segmentation, as there are multiple MRI scanner vendors and protocols.

3. Methodology

Figure 1 represents an overview of the proposed methodology for this paper and our future work. The objective of this paper is to introduce a quantitative comparison between deep learning-based methods for prostate segmentation. Our main attempt is to train the deep learning models to segment prostate cancer from MRI scans. In this stage, we selected a set of public datasets to train and test the developed models such as Promise12 [11], ISBI2013 [15], and ProstateX [31]. Our strategy depends on the use of 2D and 3D segmentation models. As several medical image analysis studies used the U-Net architecture, we train the U-Net [8] and 2D-Unet [30] models for the 2D segmentation. For 3D segmentation, we train 3DFCN [1], 3D-Unet [23], and MS-Net [27].

All the architectures chosen above are employed in the literature. However, we applied data augmentation on the MRI datasets. We employed a resizing procedure to set the same slices number for each patient. Also, we set the same dimensions for all slices. Then, we split the dataset into training and testing data. To evaluate the performance of the segmentation models, we compute the Dice coefficient and Hausdorff distance. We display the results in the format of mean+std.

It should be noted that we perform the testing of the segmentation model to delineate prostate tumors in MRI images using the ProstateX [31] dataset. This prepares for our final goal that consists of the use of the obtained segmented images with their original MRI imaging to extract a set of radiomics features. Further, we will develop a classification model that adopts the extracted radiomics to discriminate between benign or malignant tumors.

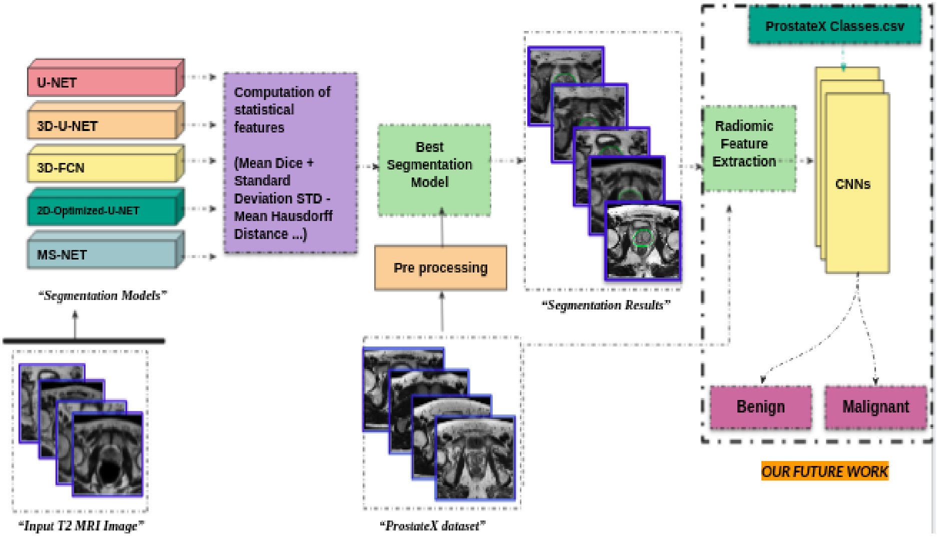


Figure 1. The schematic illustration of the proposed methodology. The first stage consists of training several deep learning-based methods for prostate delineation. Then testing the best segmentation model using the ProstateX dataset. The second stage stands for our future work that uses the segmentation results to extract a set of quantitative features in order to employ them in the classification of prostate tumors into benign or malignant.

4. Experimentation and results

4.1. Datasets

- **PROMISE12 challenge [11]** is a benchmark for evaluating segmentation algorithms of the MRI prostate. It contains a total of 50 patients transversal T2-weighted MR images of the prostate and the corresponding true mask segmentation acquired in different hospitals that and variations in voxel size, dynamic range, position, field of view, and anatomic appearance, use a variety of vendors and acquisition protocols.e
- **ISBI2013 challenge [15]** consists of 60 patient scans saved in Dicom files (1.5 MRI and T3 MRI). The scans were acquired from Radboud University Nijmegen Medical Centre [RUNMC], Netherland. The ground truth for segmentation has been created by Drs. Nicolas Bloch (Boston University School of Medicine), Mirabela Rusu (Case Western University), Drs. Henkjan Huisman, Geert Litjens, Futterer at RUNMC.
- **ProstateX Challenge [31]** focused on quantitative image analysis methods for the diagnostic classification of clinically significant prostate cancer. The largest mpMRI dataset contains 346 participants and 349 studies. It consists of a total of 204 and 140 mpMRIs for training and testing respectively. The mpMRIs dataset includes the following sequences: T2-weighted (T2), diffusion-weighted (DW), apparent diffusion coefficient (ADC) map, and K trans (computed from dynamic contrast-enhanced -DCE- T1-weighted series).

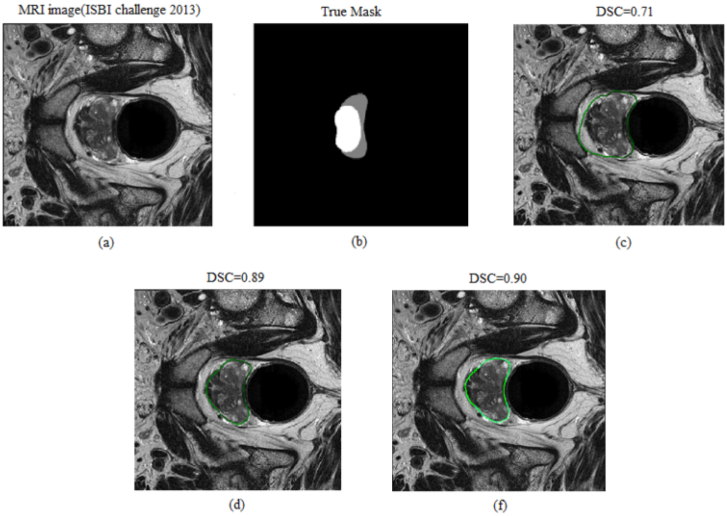
4.2. Segmentation Results

To evaluate the performance of several the segmentation models, we trained them on Promise12 [11] and ISBI2013 [15] datasets. We present in Table 1 and Table 2 the segmentation results in terms of the Dice coefficient and Hausdorff Distance.

Table 1. Comparing the performance of the segmentation models using ISBI2013 challenge dataset.

Model	Dice±STD	Hausdorff Distance±STD (mm)
MS-Net [27]	0.899 ±1.960	9.511 ±4.011
2D-Unet [30]	0.901 ±0.015	6.030 ±3.082
3D-Unet [27]	0.722 ±0.020	17.761 ±2.924

Figure 2 and Figure 3 present the qualitative results of the segmentation models with the ISBI2013 and Promise12 datasets, respectively.



2013 chllenge.png 2013 chllenge.png

Figure 2. Segmentation results on ISBI 2013 Dataset. In the top row, from left to right, the raw image, ground truth, the segmentation results of 3D-Unet, respectively. In the button row, the segmentation results of MS-Net in the left, and the segmentation results of optimized 2D-Unet in the right.

Table 2. Comparing the performance of the segmentation models using PROMISE12 challenge dataset.

Model	Dice±STD	Hausdorff Distance±STD (mm)
U-Net [8]	0.880 ±0.041	17.690 ±2.087
3D-FCN [1]	0.790 ±0.050	12.910 ±4.005
2D-Unet [30]	0.899 ±0.021	7.661 ±3.924

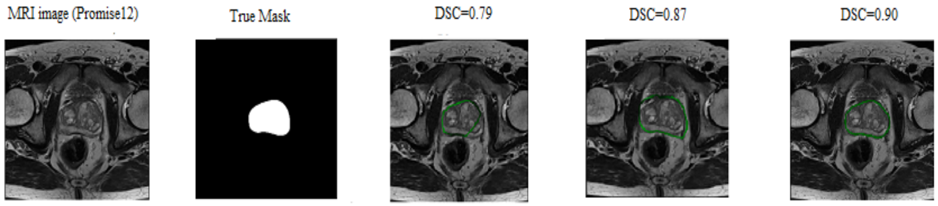


Figure 3. Segmentation results on the Promise12 dataset. From left to right, the raw image, ground truth, the segmentation results of 3D-FCN, and the segmentation results of U-Net, respectively.

The results presented in the Table 1 and Table 2 demonstrate that the 2D-optimised Unet [30] provides the best dice coefficient with the ISBI and Promise12 datasets. The 2D-Unet model achieves a Hausdorff Distance (HD) of 6.03 mm. These results demonstrate that 2D-Unet model can give accurate segmentation results on ProsateX Dataset. We show a qualitative segmentation example in Figure 4.

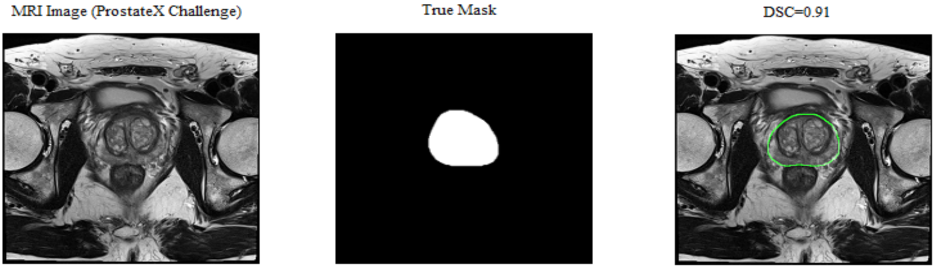


Figure 4. Segmentation results on the ProstateX dataset using 2D-Unet. From left to right, the raw image, ground truth and the segmentation result using 2D-UNet, respectively.

Table 3 presents the segmentation results when testing the segmentation models mentioned above on the ProstateX dataset.

Table 3. Quantitative segmentation result on prostatex dataset.

Model	**Dice±STD**	**Hausdorff Distance±STD (mm)**
U-Net [30]	0.791±0.151	17.020 ±2.884
3D-UNet [30]	0.701 ±0.078	18.001 ±3.108
3D-FCN [1]	0.721 ±0.047	13.411 ±5.264
2U-Net [30]	0.898±0.051	7.690 ±2.987

5. Conclusion and Future Work

This paper presented a comparative study for state-of-the-art deep learning-based segmentation models (U-net, 2D-Unet, 3D-Unet, Ms-Net and 3DFCN) for prostate delineation in MRI images. Different metrics were used to compare the prostate delineation models like the DSC coefficient and Hausdorff Distance. In future work, we will use the segmentation results to extract a set of radiomics to be inputted into a classifier to discriminate between prostate cancer classes (e.g., benign or malignant).

Acknowledgement

The Spanish Government partly supported this research through Project PID2019-105789RB-I00.

References

[1] Milletari F, Navab N, Ahmadi SA. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In2016 fourth international conference on 3D vision (3DV) 2016 Oct 25 (pp. 565-571). IEEE.

[2] Yu L, Yang X, Chen H, Qin J, Heng PA. Volumetric ConvNets with mixed residual connections for automated prostate segmentation from 3D MR images. InProceedings of the AAAI Conference on Artificial Intelligence 2017 Feb 10 (Vol. 31, No. 1).

[3] Klein S, van der Heide UA, Lips IM, van Vulpen M, Staring M, Pluim JP. Automatic segmentation of the prostate in 3D MR images by atlas matching using localized mutual information. Med Phys. 2008 Apr;35(4):1407-17. doi: 10.1118/1.2842076. PMID: 18491536.

[4] Toth R, Madabhushi A. Multifeature landmark-free active appearance models: application to prostate MRI segmentation. IEEE Trans Med Imaging. 2012 Aug;31(8):1638-50. doi: 10.1109/TMI.2012.2201498. Epub 2012 May 30. PMID: 22665505.

[5] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556. 2014 Sep 4.

[6] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. InProceedings of the IEEE conference on computer vision and pattern recognition 2015 (pp. 1-9).

[7] Meyer A, Mehrtash A, Rak M, Schindele D, Schostak M, Tempany C, Kapur T, Abolmaesumi P, Fedorov A, Hansen C. Automatic high resolution segmentation of the prostate from multi-planar MRI. In2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018) 2018 Apr 4 (pp. 177-181). IEEE.

[8] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. InInternational Conference on Medical image computing and computer-assisted intervention 2015 Oct 5 (pp. 234-241). Springer, Cham.

[9] Cheng R, Roth HR, Lu L, Wang S, Turkbey B, Gandler W, McCreedy ES, Agarwal HK, Choyke P, Summers RM, McAuliffe MJ. Active appearance model and deep learning for more accurate prostate segmentation on MRI. InMedical imaging 2016: Image processing 2016 Mar 21 (Vol. 9784, p. 97842I). International Society for Optics and Photonics.

[10] Clark T, Wong A, Haider MA, Khalvati F. Fully deep convolutional neural networks for segmentation of the prostate gland in diffusion-weighted MR images. InInternational Conference Image Analysis and Recognition 2017 Jul 5 (pp. 97-104). Springer, Cham.

[11] Litjens G, Toth R, van de Ven W, Hoeks C, Kerkstra S, van Ginneken B, Vincent G, Guillard G, Birbeck N, Zhang J, Strand R. Evaluation of prostate segmeZhu Q, Du B, Turkbey B, Choyke PL, Yan P. Deeply-supervised CNN for prostate segmentation. In2017 international joint conference on neural networks (IJCNN) 2017 May 14 (pp. 178-184). IEEE.ntation algorithms for MRI: the PROMISE12 challenge. Medical image analysis. 2014 Feb 1;18(2):359-73.

[12] Zhu Q, Du B, Turkbey B, Choyke PL, Yan P. Deeply-supervised CNN for prostate segmentation. In2017 international joint conference on neural networks (IJCNN) 2017 May 14 (pp. 178-184). IEEE.

[13] Tian Z, Liu L, Zhang Z, Fei B. PSNet: prostate segmentation on MRI based on a convolutional neural network. Journal of Medical Imaging. 2018 Jan;5(2):021208.

[14] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. InProceedings of the IEEE conference on computer vision and pattern recognition 2015 (pp. 3431-3440).

[15] Nicholas Bloch AM. NCI-ISBI 2013 Challenge: Automated Segmentation of Prostate Structures [Internet]. The Cancer Imaging Archive; 2015. Available from: https://wiki.cancerimagingarchive.net/x/B4NEAQ

[16] Karimi D, Samei G, Kesch C, Nir G, Salcudean SE. Prostate segmentation in MRI using a convolutional neural network architecture and training strategy based on statistical shape models. International journal of computer assisted radiology and surgery. 2018 Aug;13(8):1211-9.

[17] Yoo S, Gujrathi I, Haider MA, Khalvati F. Prostate cancer detection using deep convolutional neural networks. Scientific reports. 2019 Dec 20;9(1):1-0.

[18] Zhu Y, Wei R, Gao G, Ding L, Zhang X, Wang X, Zhang J. Fully automatic segmentation on prostate MR images based on cascaded fully convolution network. Journal of Magnetic Resonance Imaging. 2019 Apr;49(4):1149-56.

[19] Wang B, Lei Y, Tian S, Wang T, Liu Y, Patel P, Jani AB, Mao H, Curran WJ, Liu T, Yang X. Deeply supervised 3D fully convolutional networks with group dilated convolution for automatic MRI prostate segmentation. Medical physics. 2019 Apr;46(4):1707-18.

[20] Liu Y, Yang G, Mirak SA, Hosseiny M, Azadikhah A, Zhong X, Reiter RE, Lee Y, Raman SS, Sung K. Automatic prostate zonal segmentation using fully convolutional network with feature pyramid attention. IEEE Access. 2019 Nov 8;7:163626-32.

[21] Nie D, Shen D. Semantic-guided encoder feature learning for blurry boundary delineation. arXiv preprint arXiv:1906.04306. 2019 Jun 10.

[22] Zhu Q, Du B, Yan P. Boundary-weighted domain adaptive neural network for prostate MR image segmentation. IEEE transactions on medical imaging. 2019 Aug 13;39(3):753-63.

[23] Isensee F, Petersen J, Klein A, Zimmerer D, Jaeger PF, Kohl S, Wasserthal J, Koehler G, Norajitra T, Wirkert S, Maier-Hein KH. nnu-net: Self-adapting framework for u-net-based medical image segmentation. arXiv preprint arXiv:1809.10486. 2018 Sep 27.

[24] Simpson AL, Antonelli M, Bakas S, Bilello M, Farahani K, Van Ginneken B, Kopp-Schneider A, Landman BA, Litjens G, Menze B, Ronneberger O. A large annotated medical image dataset for the development and evaluation of segmentation algorithms. arXiv preprint arXiv:1902.09063. 2019 Feb 25.

[25] Rundo L, Han C, Nagano Y, Zhang J, Hataya R, Militello C, Tangherloni A, Nobile MS, Ferretti C, Besozzi D, Gilardi MC. USE-Net: Incorporating Squeeze-and-Excitation blocks into U-Net for prostate zonal segmentation of multi-institutional MRI datasets. Neurocomputing. 2019 Nov 6;365:31-43.

[26] https://www.overleaf.com/project/607c494fe8d9d7b627cd6869 Zavala-Romero O, Breto AL, Xu IR, Chang YC, Gautney N, Dal Pra A, Abramowitz MC, Pollack A, Stoyanova R. Segmentation of prostate and prostate zones using deep learning. Strahlentherapie und Onkologie. 2020 Oct;196(10):932-42.

[27] Liu Q, Dou Q, Yu L, Heng PA. MS-net: Multi-site network for improving prostate segmentation with heterogeneous MRI data. IEEE transactions on medical imaging. 2020 Feb 17;39(9):2713-24.

[28] Wright, L., 2019. Ranger deep learning optimizer. URL: https://github.com/lessw2020/Ranger-Deep-Learning-Optimizer.

[29] Misra D. Mish: A self regularized non-monotonic activation function. arXiv preprint arXiv:1908.08681. 2019 Aug 23.

[30] Gillespie D, Kendrick C, Boon I, Boon C, Rattay T, Yap MH. Deep learning in magnetic resonance prostate segmentation: A review and a new perspective. arXiv preprint arXiv:2011.07795. 2020 Nov 16.

[31] Geert Litjens, Oscar Debats, Jelle Barentsz, Nico Karssemeijer, and Henkjan Huisman. ”ProstateX Challenge data”, The Cancer Imaging Archive (2017). DOI: 10.7937/K9TCIA.2017.MURS5CL

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210149

Segmenting the Optic Disc Using a Deep Learning Ensemble Model Based on OWA Operators

Mohammed Yousef Salem ALI [a,1], Mohamed ABDEL-NASSER [a,c]
Mohammed JABREEL [b] Aida VALLS [a] and Marc BAGET [d]

[a] *Departament Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili, Tarragona (Catalonia), Spain*
[b] *Microsoft Advanced Technology Lab, Cairo, Egypt*
[c] *Department of Electrical Engineering, Aswan University, 81528 Aswan, Egypt*
[d] *IISPV, Hospital Universitari Sant Joan de Reus, Spain*

Abstract. The optic disc (OD) is the point where the retinal vessels begin. OD carries essential information linked to Diabetic Retinopathy and glaucoma that may cause vision loss. Therefore, accurate segmentation of the optic disc from eye fundus images is essential to develop efficient automated DR and glaucoma detection systems. This paper presents a deep learning-based system for OD segmentation based on an ensemble of efficient semantic segmentation models for medical image segmentation. The aggregation of the different DL models was performed with the ordered weighted averaging (OWA) operators. We proposed the use of a dynamically generated set of weights that can give a different contribution to the models according to their performance during the segmentation of OD in the eye fundus images. The effectiveness of the proposed system was assessed on a fundus image dataset collected from the Hospital Sant Joan de Reus. We obtained Jaccard, Dice, Precision, and Recall scores of 95.40, 95.10, 96.70, and 93.90%, respectively.

Keywords. Fundus image, Optic disc segmentation, Deep learning, OWA operators

1. Introduction

Retinal diseases like diabetic retinopathy (DR) and glaucoma highly affect the optic disc (OD) of the human eye–OD is also called the optic nerve head. The early detection and treatment of DR and glaucoma are essential to prevent total vision loss [1]. Ophthalmologists inspect eye fundus images to detect the signs of such diseases. Yet, the manual analysis of many fundus images is expensive in terms of both time and effort. Modern computer-aided diagnosis (CAD) systems can analyze the medical images and provide a diagnosis as accurate as ophthalmologists with many years of experience [2]. Deep learning (DL) technologies have become the cornerstone of several modern CAD systems.

[1] Corresponding Author: Mohammed Yousef Salem Ali. E-mail: horbio10@gmail.com

Recently, several DL-based automated systems have been proposed to segment the OD automatically. Most of them use convolutional neural networks (CNNs) to automatically learn representative and high-level features from the input fundus images to achieve accurate segmentation [3]. For instance, the authors of [4] modified the U-Net model by reducing the size of the filters in all CNN layers to segment the OD directly. They stated that this modification produced a lighter model and improved the segmentation performance. With the DRIONS-DB dataset, they achieved IOU and Dice scores of 89 and 94%, respectively. With RIM-ONE v.3, they obtained IOU and Dice scores 89 and 95%, respectively. The authors of [5] modified the U-Net and DeepLabv3+ models by inserting an attention module between the encoder and decoder networks and adding a conditional random field layer in the output layer. The method was evaluated using DRIONS-DB, RIM-ONE v.3, and DRISHTI-GS fundus image datasets. They achieved a 95% Dice and 91% Jaccard with DRIONS-DB, a 97% Dice and a 94% Jaccard with RIM-ONE, and a 96% Dice and a 92%Jaccard with DRISHTI-GS.

In [6], a multi-label deep neural network called M-Net was proposed for segmenting OD and optic cup (OC) jointly. M-Net includes a multi-scale input layer, U-shape CNN, a side-output layer, and a multi-label loss function based on the dice loss. The side-output layer works as an initial classifier to provide a local prediction map for different scale layers. M-Net achieved an accuracy of 98.30% with the ORIGA dataset. The study confirmed that the accurate segmentation process of OD and OC jointly is essential to glaucoma detection and diagnosis. The authors of [7] leveraged the multi-task learning and dense connections to segment the OD and OC jointly. They evaluated their work on three datasets: DRISHTI-GS, RIM-ONE, and REFUGE. With the OD segmentation task, they achieved a 91.83% Jaccard and a 95.97% Dice with DRISHTI-GS, a 91.01% Jaccard and a 95.82% Dice with RIM-ONE, and an 88.37% Jaccard with REFUGE.

Although the methods mentioned above achieved good OD segmentation accuracy, no individual model performs best under all conditions. The fusion of the predictions of individual models can improve the segmentation accuracy [8]. In [9], a CNN model was trained using probability masks instead of binary masks to segment OD and OC. Each probability mask was obtained by fusing segmentation masks made by multiple experts.

In [10], an ensemble of different CNN architectures for medical image classification is used. The study demonstrated that different CNNs learn different levels of semantic image representation, and therefore an ensemble of CNNs produces richer features. [11] fused the predicted labels of five CNNs models (Resnet50, Inceptionv3, Xception, Dense121, and Dense169) to improve the results of DR classification in fundus images.

However, most existing methods use fixed weights associated with each model to construct the ensemble. This strategy may not be appropriate for all situations. Differently, in this paper we employ the ordered weighted averaging operator (OWA) [12] to aggregate the information provided by several OD segmentation models. More specifically, this paper proposes an ensemble-based OD segmentation system based on a set of deep learning models and an aggregation stage based on OWA.

OWA is an aggregation operator that uses a set of weights to define the aggregation policy for the inputs, which can vary from full conjunctiveness to full disjunctiveness. These weights do not indicate the importance of each source of information but the importance of the input values that are merged. The definition of the proper weights is a key issue in this aggregation operator. A dynamic way of generating weights is through quantifier functions [12]. Notably, this OWA-based ensemble of DL models will allow the construction of a dynamic OD segmentation system.

The remainder of this paper is organized as follows. Section 2 explains the proposed OD segmentation system. Section 3 presents the experiments and results. Finally, the conclusion and future work are provided in section 4.

2. The Proposed Optic Disc Segmentation Method

To develop a reliable and efficient OD segmentation system, we first train several deep learning segmentation models. Figure 1 shows the procedure of segmenting a new image. The test image fed into *N* individual OD segmentation models. Then, the OWA operator aggregates the different segmentation masks given by the models to produce the final segmentation mask (final prediction).

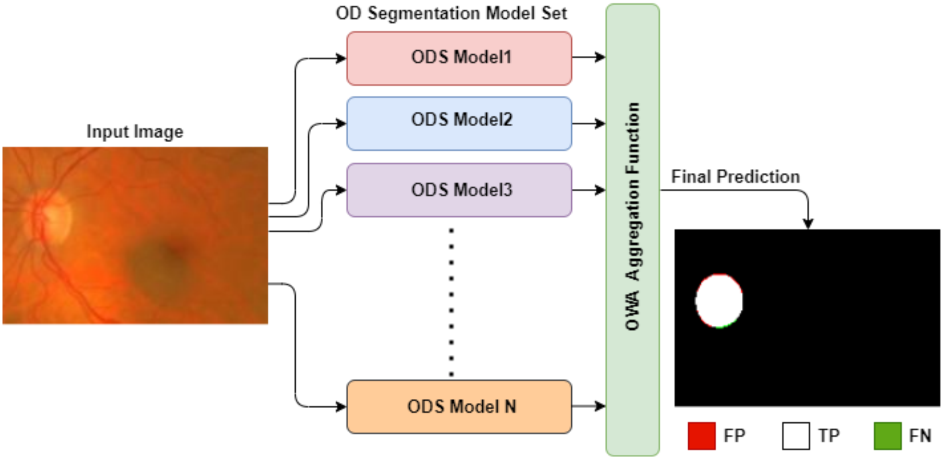


Figure 1. Structure of the proposed method.

2.1. Constructing individual OD segmantation models

In this study, we developed ten OD segmentation models based on different state-of-the-art deep learning-based semantic segmentation models. Specifically, we used the Unet, Gated Skip Connections (GSCs), DoubleU-Net, DeepLabV3+, CGNet, ERFNet, SegNet, ESNet, LinkNet, and SQNet models. Each of them was trained to maximize the segmentation accuracy. The architecture of all models based on the encoding and decoding method for work except the CGNet. The binary cross-entropy loss function and ADAM optimizer were used to train all models. The source code of CGNet, ERFNet, SegNet, ESNet, LinkNet, and SQNet can be found at `https://github.com/xiaoyufenfei/Efficient-Segmentation-Networks/blob/master/model`. We briefly introduce each model below.

- **UNet [13]** had five blocks in the contracting path (downsampling path) and five blocks in expanding path (upsampling path). The model contained a total of twenty-three convolutional layers. Source code available at https://github.com/jakeret/unet/blob/master/src/unet/unet.py

- **Gated Skip Connections (GSCs) [14]** is a modified version of UNet. It had five encoder blocks with two convolutional layers, five decoder blocks with four convolutional layers, a total of thirty-three convolutional layers, and a gated skip connection mechanism in each decoder block with a convolutional layer.
- **DoubleU-Net [15]** included two stacked U-Net [13] in sequence. The first stack based on pre-trained VGG19 of an encoder with four blocks in each encoder-decoder, Second stack similar Unet with four blocks encoder-decoder, atrous spatial pyramid pooling after per encoder for each stack. Each block contained two convolutional layers and max-pooling. Source code available at https://github.com/DebeshJha/2020-CBMS-DoubleU-Net
- **DeepLabV3+ [16]** included ResNet50 backbone, deep separation convolution, and atrous separable convolution. Source code available at https://github.com/srihari-humbarwadi/DeepLabV3_Plus-Tensorflow2.0/blob/master/deeplab.py
- **CGNet [17]** included three down-sampling stages with Context Guided (CG) blocks for each stage. In total, it contained a total of fifty-one convolutional layers with no decoder modules.
- **ERFNet [18]** included two types of residual layer design (bottleneck and non-bottleneck), skip connections between encoder and decoder, contained a total of twenty-three convolutional layers.
- **SegNet [19]** had four blocks in the encoder part with thirteen convolutional layers of the VGG16 backbone network, four blocks in the decoder part with thirteen convolutional layers too.
- **ESNet [20]** had three blocks in the each encoder and decoder part,contained factorized convolution unit and parallel factorized convolution unit, contained a total of eighteen convolutional layers.
- **LinkNet [21]** had four blocks in each encoder and decoder part, four convolutional layers per each encoder block, residual block per two layers in each encoder block and two convolutional layers per each decoder block (26 layers).
- **SQNet [22]** is based on a modified SqueezeNet 1.1 (fire module). It contained eight SqueezeNet with three convolutional operations in each fire module, four parallel dilated convolutions and a refinement module in the decoder network. Exponential Linear Units were used in SQNet to avoid the bias shift.

2.2. OWA-based Aggregation Function

For the aggregation step, we propose the use of the Ordered Weighted Average operator defined by Yager [12]. This operator performs a weighted average of a vector of input values $X = (x_1, x_2, ...x_m)$, but before averaging, the values are sorted from the best to the worst (denoted with σ). In that way, the weighting vector W is associating importance to each position in the ordered input vector, regardless of the source of the value in that position. This is expressed in (Equation 1). Notice that the sum of weights must be equal to 1.

$$F_W\left(x_1, x_2, \ldots, x_m\right) = \sum_{k=1}^{m} w_k x_{\sigma_k} \tag{1}$$

The key property of this operator is that the weights can model different aggregation policies, ranging from situations of full andness (when $w_m = 1$ and rest are 0) to full orness (when $w_1 = 1$ and rest are 0). Any possibility in between can be represented with an appropriate combination of the weights.

Manually setting the weights for a certain policy may be difficult. One usual way of generating the weighting vector automatically is through Regular Increasing Monotone fuzzy linguistic quantifiers [12]. Each weight is then obtained using the expression in (Equation 2), where $Q(z)$ is a function that corresponds to a fuzzy quantifier. This quantifier can generate weights representing different ”quantities” of agreement, which is a factor to determine the degree of andness/orness of the aggregation.

$$w_k = Q\left(\frac{k}{m}\right) - Q\left(\frac{k-1}{m}\right) \tag{2}$$

Three common aggregation strategies that can be defined using a fuzzy quantifier are the following: i) *At least half*, Q^1, which is more tolerant and focuses only on a small subset of the best values aggregated (Equation 3). It corresponds to a situation of low andness. ii) *As many as possible*, Q^2, which is quite strict in the aggregation as it gives most importance to the worst values aggregated (Equation 4). It represents a situation of high andness. iii) *Most*, Q^3, which represents a case of finding the majority value (Equation 5). It corresponds to a situation of moderate andness (close to neutrality).

$$Q^1(z) = \begin{cases} 2z & \text{if } \; 0 \leq z \leq 0.5 \\ 1 & \text{if } \; 0.5 < z \leq 1 \end{cases} \tag{3}$$

$$Q^2(z) = \begin{cases} 0 & \text{if } \; 0 \leq z \leq 0.5 \\ 2z - 1.0 & \text{if } \; 0.5 < z \leq 1 \end{cases} \tag{4}$$

$$Q^3(z) = \begin{cases} 0 & \text{if } \; 0 \leq z \leq 0.3 \\ 2(z - 0.3) & \text{if } \; 0.3 < z \leq 0.8 \\ 1 & \text{if } \; 0.8 < z \leq 1 \end{cases} \tag{5}$$

These three quantifiers are the ones that have been used for aggregating the probability values obtained with the different DL models for each class. The aggregation is applied at each pixel of the image studied. A threshold of 0.5 is applied to the result and the class with the maximum activation is taken as label.

2.3. Evaluation Metrics

The evaluation is done with the segmented masks and the corresponding ground truth. The segmented masks were binarized so that pixels in the OD have the label ’1’ while pixels in the background have the label ’0’. We computed the number of true positive (TP), true negative (TN), false positive (FP), and false-negative (FN) pixels.

Then, we computed the evaluation metrics commonly used in the OD segmentation task: the intersection-over-union (IOU), Precision (Prec.), Recall, and Dice coefficient (Dice).

i) *IOU* is the ratio of the intersection between the two masks (original and predicted mask) for OD with respect to their union. It is computed using Equation 6.

$$IOU(A,B) = \frac{A \cap B}{A \cup B} = \frac{TP}{TP+FP+FN} \tag{6}$$

ii) *Precision* is the ratio of TP and the pixels predicted as OD.

$$\text{Precision } = \frac{TP}{TP+FP} \tag{7}$$

iii) *Recall* is the ratio of TP and the correct pixels in OD according to ground truth.

$$\text{Recall} = \frac{TP}{TP+FN} \tag{8}$$

iv) *Dice coefficient* makes an harmonic mean of precision and recall. It is also referred to as F-score.

$$\text{Dice } = 2 \cdot \frac{\text{Precision. Recall}}{\text{Precision } + \text{ Recall}} \tag{9}$$

3. Results and Discussions

3.1. Dataset and Experimental Setup

Here, we used an in-house fundus image dataset collected from Hospital Universitari Sant Joan de Reus (SJR). It includes 105 images that were manually labeled by ophthalmologists in SJR. The original size of the images is 3008 × 2000. We randomly divided the dataset into three subsets: the training set (56 images), the validation set (14 images), and the testing set (35 images). We used data augmentation techniques to increase the training data set size, generating a training set with 1120 images. Then, we resized the images and the ground truths to 256 × 384 to make the training process faster. The validation set was used to save the best checkpoint of the trained models. We employed the binary cross-entropy loss function in all models. We trained the models with 50 epochs using Adam optimizer with a learning rate of 0.001, and a batch size of 4 images.

3.2. Analysis of the Results of Individual DL models

We trained and tested 10 state-of-the-art deep CNN independently for OD image segmentation on the SJR dataset. The results are shown in Table 1. We noted that the GSCs model has the best result of IOU and Dice. Then ERFNet gives the best result of the Precision metric. Also, the DoubleU-Net has the best results of the Recall metric. So, we can see that there is not a unique winner method in all quality indices. This fact motivates the idea of using an ensemble of a subset of these models. To illustrate the results, we displayed some sample masks obtained for the three best IOUs OD segmentation models

Table 1. Performance comparison on SJR dataset.

Models	Evaluation Metrics			
	IOU (%)	**Dice (%)**	**Prec.(%)**	**Recall (%)**
ODS1: GSCs	**95.1**	**94.9**	97.0	92.9
ODS2: DoubleU-Net	94.4	94.2	93.3	**95.0**
ODS3: DeepLabV3+	94.1	93.8	97.7	90.1
ODS4: U-Net	92.4	91.8	96.6	87.5
ODS5: LinkNet	92.4	91.8	99.7	85.3
ODS6: SQNet	91.9	91.2	97.6	84.8
ODS7: ESNet	91.9	91.2	99.5	84.3
ODS8: CGNet	91.6	90.8	99.6	83.7
ODS9: ERFNet	91.5	90.7	**99.8**	83.4
ODS10: SegNet	90.5	89.4	99.3	81.8

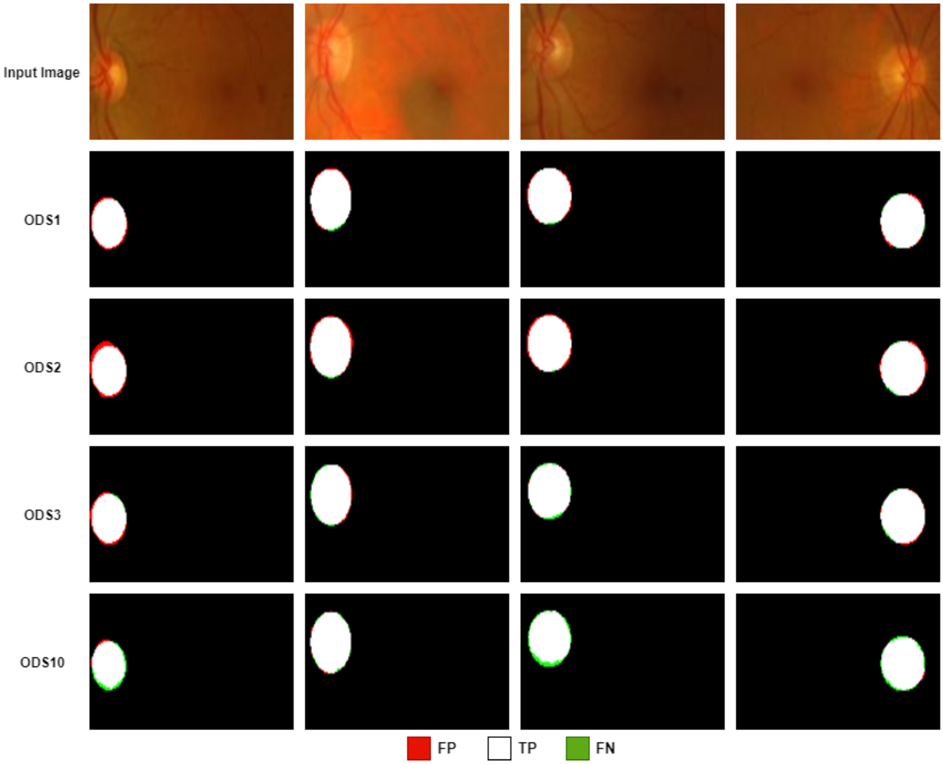


Figure 2. Results of the best three OD segmentation models and the worst one. Here ODS1, ODS2, ODS3, ODS10 indicate to GSCs, DoubleU-Net, DeepLabV3+, SegNet models.

and the worst one in Figure 2. Pixels in red are the FP, in green the FN, while white ones TP are the correctly classified.

3.3. *Analysis of the Results of the OWA-based DL Ensemble System*

In this section, we compared the three OWA aggregation policies defined before in

Section 2: Q^1 uses a disjunctive approach, considering only the models with the best performance, Q^2 applies a strong conjunctive approach, requiring that most models agree on the same output, while Q^3 takes an intermediate point between the previous ones.

We used different subsets of models to build the ensemble. Reducing the number of models is interesting to decrease the execution time. After ranking the DL models with the IOU index, we compared a set with the Top-7 models: GSCs, DoubleU-Net, DeepLabV3+, U-Net, LinkNet, SQNet, and ESNet; a set with the Top-5 models: GSCs, DoubleU-Net, DeepLabV3+, U-Net, and LinkNet; and with the Top-3 models: GSCs, DoubleU-Net, and DeepLabV3+. Results are shown in Table 2.

Table 2. Results of different OWA's policies with a different number of aggregated models.

Aggreg. Mod.	**Evaluation Metrics of OWA Policies (%)**											
	At least half, Q^1				**As many as possible, Q^2**				**Most, Q^3**			
	IOU	**Dice**	**Pre**	**Rec**	**IOU**	**Dice**	**Pre**	**Rec**	**IOU**	**Dice**	**Pre**	**Rec**
10 Mod.	93.4	92.9	**99.6**	87.3	93.7	93.2	**99.5**	87.9	93.5	93.0	**99.6**	87.5
Top-7	94.1	93.8	99.2	89.2	94.4	94.1	99.0	89.8	94.3	94.0	99.1	89.6
Top-5	94.9	94.6	98.3	91.4	95.1	94.8	98.3	91.9	95.0	94.7	98.2	91.7
Top-3	**95.2**	**95.0**	97.4	**93.0**	**95.4**	**95.1**	96.7	**93.9**	**95.3**	**95.1**	97.2	**93.3**

With the three aggregation policies, we can see that performance indicators are better when we reduce the number of models in the ensemble. The best results are obtained with the Top-3 ensemble, which achieved an IOU of 95.4% and Dice of 95.1%. We observe that the IOU and Dice scores of the aggregated models are better than the ones of the individual models. However, precision is higher when using more models. An improvement in recall means that we reduce the False Negatives. The non-improvement of precision means that the number of FPs is the same or increase a bit. However, as IOU and Dice are better it means that the improvement in FNs is considerably larger than the increase of FPs.

Regarding OWA, we can also observe that the most conjunctive policy (i.e. as many as possible, Q^2) is outperforming the others. Comparing the results with the Q^2 ensemble system and the individual models, we observe an improvement in the performance (Table 1). In this experiment, results improve when using a strict conjunctive aggregation policy, which is the one that requires a high agreement of the models in order to choose the final class. As OWA is using the best performing subset of n models, this aggregation is taking into consideration the consensus of the n models regarding the class probability of each pixel (i.e. probability of each class given by each model). As this consensus may be difficult to find, it is reasonable that smaller subsets perform better than adding more models with fewer performance indicators. This observation is interesting because it means that it is not necessary to include a large number of DL models in the ensemble. From the improvement with respect to individual models, we can conclude that the ensemble approach can improve the segmentation result, correcting some of the mistakes of individual models. Figure 3 displays the segmentation made with the three different OWA policies when using the Top-3 methods with the same images than previous figure.

It is interesting to determine the statistical significance of the differences in performance between the proposed ensemble and GSCs (the best individual model) in terms of the IOU and Dice. To do so, we used Student's t-test (significance level < 0.05) to determine the difference in IOU and Dice values. The p-values obtained are higher than 0.05, indicating no statistical significance.

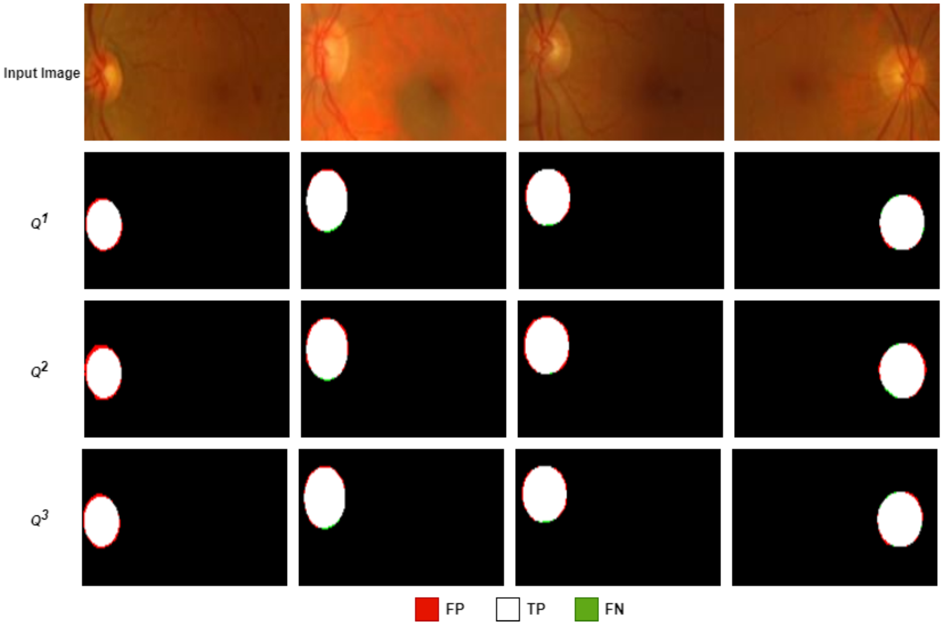


Figure 3. Results of the three policies of the OWA proposed method. Here Q1, Q2, Q3 indicate to the policies

4. Conclusions and future work

This paper presented an automated system for segmenting the OD in fundus images of the human eye based on accurate deep learning-based segmentation models and OWA operators. We assessed the performance of three different policies and a different number of individual OD segmentation models. We found that a high conjunctive aggregation with a reduced subset of the classifiers leads to the best OD segmentation model. With an in-house fundus images dataset collected from the Hospital Sant Joan de Reus, our system achieved an IoU and Dice scores higher than 95%, a precision higher than 96%, and a recall higher than 93%. Using an ensemble of DL models increases the training time significantly because several individual OD segmentation models must be built before constructing the ensemble. The fact that a combination of only three models has given the best results makes this ensemble approach feasible. The inference time is low enough to be used in a hospital in real-time. Moreover, the improvement achieved with the proposed ensemble system can give a more accurate segmentation of the OD, which reduces human errors for ophthalmologists in detecting eye pathologies. Future work will include using the proposed OD segmentation model to develop detection tools for eye diseases like glaucoma or diabetic retinopathy.

Acknowledgements

This work has been funded by the research project PI18/00169 from Instituto de Salud Carlos III & FEDER funds. The University Rovira i Virgili also supports this work with project 2019PFR-B2-61.

References

[1] Mary VS, Rajsingh EB, Naik GR. Retinal fundus image analysis for diagnosis of glaucoma: a comprehensive survey. IEEE Access. 2016;.

[2] Jani K, Srivastava R, Srivastava S, Anand A. Computer aided medical image analysis for capsule endoscopy using conventional machine learning and deep learning. In: 2019 7th International Conference on Smart Computing & Communications (ICSCC). IEEE; 2019. p. 1–5.

[3] Fourcade A, Khonsari R. Deep learning in medical image analysis: A third eye for doctors. Journal of stomatology, oral and maxillofacial surgery. 2019;120(4):279–288.

[4] Sevastopolsky A. Optic disc and cup segmentation methods for glaucoma detection with modification of U-Net convolutional neural network. Pattern Recognition and Image Analysis. 2017;27(3):618–624.

[5] Bhatkalkar BJ, Reddy DR, Prabhu S, Bhandary SV. Improving the performance of convolutional neural network for the segmentation of optic disc in fundus images using attention gates and conditional random fields. IEEE Access. 2020;8:29299–29310.

[6] Fu H, Cheng J, Xu Y, Wong DWK, Liu J, Cao X. Joint optic disc and cup segmentation based on multilabel deep network and polar transformation. IEEE transactions on medical imaging. 2018;37(7):1597–1605.

[7] Tabassum M, Khan TM, Arsalan M, Naqvi SS, Ahmed M, Madni HA, et al. CDED-Net: Joint segmentation of optic disc and optic cup for glaucoma screening. IEEE Access. 2020;8:102733–102747.

[8] Dietterich TG. Ensemble Learning, The Handbook of Brain Theory and Neural Networks, MA Arbib. Cambridge, MA: MIT Press; 2002.

[9] Mangipudi PS, Pandey HM, Choudhary A. Improved optic disc and cup segmentation in Glaucomatic images using deep learning architecture. Multimedia Tools and Applications. 2021;p. 1–21.

[10] Kumar A, Kim J, Lyndon D, Fulham M, Feng D. An ensemble of fine-tuned convolutional neural networks for medical image classification. IEEE journal of biomedical and health informatics. 2016;21(1):31–40.

[11] Qummar S, Khan FG, Shah S, Khan A, Shamshirband S, Rehman ZU, et al. A deep learning ensemble approach for diabetic retinopathy detection. IEEE Access. 2019;7:150530–150539.

[12] Yager RR. Quantifier guided aggregation using OWA operators. International Journal of Intelligent Systems. 1996;11(1):49–73.

[13] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. Springer; 2015. p. 234–241.

[14] Jabreel M, Abdel-Nasser M. Promising crack segmentation method based on gated skip connection. Electronics Letters. 2020;56(10):493–495.

[15] Jha D, Riegler MA, Johansen D, Halvorsen P, Johansen HD. Doubleu-net: A deep convolutional neural network for medical image segmentation. In: 2020 IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS). IEEE; 2020. p. 558–564.

[16] Chen LC, Zhu Y, Papandreou G, Schroff F, Adam H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV); 2018. p. 801–818.

[17] Wu T, Tang S, Zhang R, Cao J, Zhang Y. Cgnet: A light-weight context guided network for semantic segmentation. IEEE Transactions on Image Processing. 2020;30:1169–1179.

[18] Romera E, Alvarez JM, Bergasa LM, Arroyo R. Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. IEEE Transactions on Intelligent Transportation Systems. 2017;19(1):263–272.

[19] Badrinarayanan V, Kendall A, Cipolla R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence. 2017;39(12):2481–2495.

[20] Wang Y, Zhou Q, Xiong J, Wu X, Jin X. Esnet: An efficient symmetric network for real-time semantic segmentation. In: Chinese Conference on Pattern Recognition and Computer Vision (PRCV). Springer; 2019. p. 41–52.

[21] Chaurasia A, Culurciello E. Linknet: Exploiting encoder representations for efficient semantic segmentation. In: 2017 IEEE Visual Communications and Image Processing (VCIP). IEEE; 2017. p. 1–4.

[22] Treml M, Arjona-Medina J, Unterthiner T, Durgesh R, Friedmann F, Schuberth P, et al. Speeding up semantic segmentation for autonomous driving. In: MLITS, NIPS Workshop. vol. 2; 2016. .

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210150

Lesion Detection in Breast Tomosynthesis Using Efficient Deep Learning and Data Augmentation Techniques

Loay HASSAN [a,1], Mohamed ABDEL-NASSER [a], Adel SALEH [b] and Domenec PUIG [a]

[a] *Department of Computer Engineering and Mathematics, Universitat Rovira i Virgili, 43007 Tarragona, Spain*
[b] *Gaist Solutions Ltd., Skipton BD23 2TZ, UK*

Abstract. Digital breast tomosynthesis (DBT) is one of the powerful breast cancer screening technologies. DBT can improve the ability of radiologists to detect breast cancer, especially in the case of dense breasts, where it beats mammography. Although many automated methods were proposed to detect breast lesions in mammographic images, very few methods were proposed for DBT due to the unavailability of enough annotated DBT images for training object detectors. In this paper, we present fully automated deep-learning breast lesion detection methods. Specifically, we study the effectiveness of two data augmentation techniques (channel replication and channel-concatenation) with five state-of-the-art deep learning detection models. Our preliminary results on a challenging publically available DBT dataset showed that the channel-concatenation data augmentation technique can significantly improve the breast lesion detection results for deep learning-based breast lesion detectors.

Keywords. Digital breast tomosynthesis, Lesion detection, CAD systems, Deep learning

1. Introduction

Worldwide, breast cancer is one of the most common cancers in women and a leading cause of death [1]. The earlier breast cancer gets detected, the better of getting successful treatment. Mammography (X-ray images of the breast) and breast ultrasonography (BUS) are the most common diagnostic tools to detect breast cancer. However, the sensitivity and specificity of breast lesion detection in mammographic and BUS images are often limited by the presence of high breast density (dense fibro glandular tissue in the breast). The main reason for this limitation is the 2-D imaging modality of these images [2].

Digital breast tomosynthesis (DBT)—a new 3-D breast cancer screening technique—has been introduced to reduce the effects of the superposition and overlapping problem of dense tissues in mammographic images by providing depth information, which yields

[1] Corresponding Author; E-mail: loay.abdelrahimosmanhassan@urv.cat.

enhanced breast lesion detection rate [3]. In DBT, a sequence of mammographic images is acquired by moving the X-ray tube in an arc over a stationary compressed breast at multiple angles [4]. These individual mammographic images are then reconstructed into a series of 3-D and high-resolution slices by computer software. These 3-D images significantly reduce the effect of dense tissues that can hide breast lesions or make it difficult to be detected [5].

Although DBT has the ability to address the limitation of tissue superimposition in mammography [4] by providing superior tissue visualization, radiologists are required to examine a greater number of slices per breast volume, which creates clinical workflow challenges. Furthermore, as the number of slices to evaluate grows, physicians' oversight of findings increases. As a result, computer-aided detection (CAD) is necessary for clinical DBT and offers a larger clinical role in increasing work performance than traditional digital mammography. It is also likely that CAD will perform better with DBT images than with mammographic images due to the higher mass margin visibility [6].

Traditional CAD systems can lead to drawbacks as it relies on hand-engineered features. By contrast, the development of deep learning-based CAD systems, which relies instead on learning the features and classification decisions end-to-end, has been demonstrated to outperform traditional computer aid software in many computer vision problems. Deep learning has gained popularity in biomedical image analysis with a promising results in cancer classification [7] and lesions detection [8]. Practical experiments show that deep learning algorithms are robust and more accurate across datasets.

Recently, deep learning has been employed in DBT images to improve Lesion or cancer detection accuracy. Many researchers have employed several deep learning architectures to identify and classify breast cancer [9]. Fan et. al [10] introduced a deep learning framework based on a 3-D version of the Mask-RCNN model for mass detection and segmentation in DBT images. The main idea of this study is to use ResNet-Feature Pyramid Network (ResNet-FPN) as a backbone for their model. The DBT slices of the same image were passed into the mask-RCNN model one by one to generate bounding boxes for each slice with a confidence score of detected mass. These bounding boxes of a breast mass from the same set of DBT images are combined, if they have overlapping ratios (i.e. intersection over union ratios) greater than 0.5.

In [11], Lai et. al. presented a DBT mass automatic segmentation algorithm using a U-Net architecture. Their method consists of six stages: data pre-processing, patch extraction, data augmentation, U-Net segmentation, voting stage, and post-processing. In data pre-processing, a top-hat transform was applied to enhance the contrast between tumor location regions and background tissues. Images were split into patches and these patches were rotated by 90 degrees to increase the number of data for patch extraction and data augmentation steps. U-Net model with 23 layers was used to perform an end-to-end pixel-wise segmentation. For the voting stage, a probabilistic prediction was made using the U-Net segmentation model for each slice. These predictions are then fused into the final image label using one of multiple voting schemes. Finally, they imposed volumetric constraints by removing clusters in the segmentation obtained by the U-Net to deal with small clusters that were mistakenly classified as breast masses.

Lotter et. al. [12] proposed a three-stage annotation-efficient deep learning model for breast cancer detection in mammographic and DBT images. In the first stage, a pre-trained ResNet model was trained for lesion classification. This trained ResNet then is used as a backbone model for RetinaNet which is trained for lesion detection as a second stage. In the third stage, the ROI extracted from the second stage is used to classify optimized 2-D images condensed from DBT.

Buda et. al [13] developed a single-stage deep learning model to detect masses and architectural distortions in DBT images. The proposed model employs YOLO for 2-D object detection with DenseNet. In this work, different loss functions have been tested including binary cross-entropy, weighted binary cross-entropy, focal loss, and reduced focal loss. The results showed that their model performed best using focal loss.

Breast cancer detection in mammographic and DBT images is still a challenging task due to variability in breast lesion shapes and sizes and breast density. However, many automated lesion detection approaches were proposed in the literature to accurately detect breast cancer in mammographic images, very few approaches for DBT have been presented due to a lack of enough annotated DBT images for training object detectors. In this paper, we develop five automated deep learning-based breast lesion detection models for DBT images based on robust object detection models like YOLO [14] and Faster R-CNN [15]. Additionally, we investigate the impact of two data augmentation techniques called channel-replication and chancel-concatenation in improving the breast lesion detection results of deep learning models.

The remainder of this paper is designed as follows. Section 2 describes the proposed detection system. The experimental results and discussion are presented in section 3. Section 4 concludes the paper and provided some lines for future work.

2. Proposed Lesion Detection System

Figure 1 shows the proposed lesion detection method. The key elements of the proposed method are data augmentation, deep learning-based detector, and non-maximum suppression (NMS). Below, we describe the proposed method in detail.

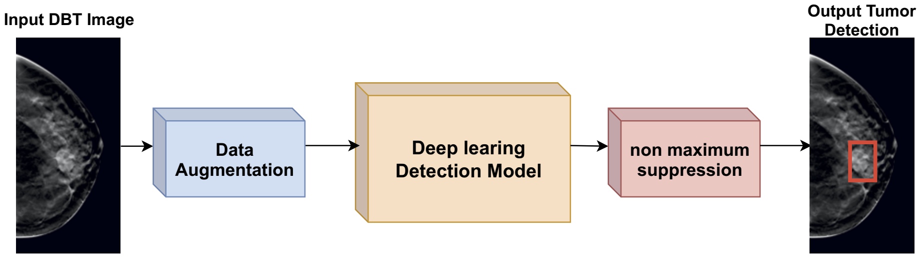


Figure 1. Lesion detection in breast tomosynthesis images based on deep learning models and robust data augmentation techniques.

2.1. *Data augmentation*

In data augmentation techniques, the number of training images is increased with different image manipulation algorithms. We use two different data augmentation techniques in this study.

- **Augmentation technique 1 (Aug1): channel replication.** We flipped all images in the training set horizontally, then for each original and flipped image I_s we apply gamma correction to adjust the overall brightness of an image using Equation 1.

$$I_\gamma = 255 \times (\frac{I_s}{255})^\gamma \tag{1}$$

Where I_γ is the output image for gamma correction and γ is the gamma correction factor.
Also, we apply the contrast limited adaptive histogram equalization (CLAHE) to enhance the image Local Contrast [16]. To calculate the clip limit for the CLAHE algorithm, we used Equation 2.

$$cliplimit = \frac{W \times H}{L}\left(1 + \frac{\alpha}{100}\left(S_{\max} - 1\right)\right) \tag{2}$$

Where $W \times H$ is the number of pixels in each histogram calculated region, L is the number of gray-scales, α is a clip factor, and S_{max} is the maximum allowable slope. However, S_{max} should set to four for still X-ray images.

- **Augmentation technique 2 (Aug2): channel-concatenation.** We used a new 3-channel data augmentation method by concatenating the original image with two post-processed images as proposed in [17]. With this augmentation technique, rather than concatenating the three gray-scale images, we used two filtered images (I_γ with $\gamma = 0.5$ and I_{clahe} with $\alpha = 1$) to concatenate with the original gray-scale image I_g. So, for each image in the training set, we produce a new 3-channel Image I as shown in Equation 3.

$$I = Concat(I_g, I_\gamma, I_{clahe}) \tag{3}$$

Here, I,I_g,I_γ, and I_{clahe} is output image, image after gamma correction and image after CLAHE equalization, respectively. Figure 2 shows sample visualization different data augmentation techniques.

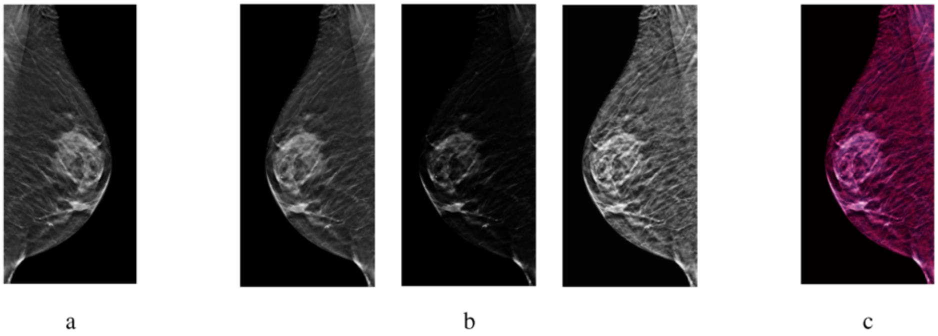


Figure 2. Samples for different data augmentation techniques. a. The original image, b. Outputs of Aug1 and c. The output of Aug2.

2.2. Individual deep learning based detection models

To build the individual deep learning-based detection models, we employed two widely used and efficient deep learning-based object detectors: YOLO and Faster-RCNN.

YOLO [14] is an object detection model that implements all of the detection stages to detect an object using a single neural network. The YOLO algorithm has obtained impressive specifications that outperform the leading detection algorithms in terms of both speed and accuracy for detecting and determining object location. The main idea of the YOLO algorithm is to apply a single neural network to the full image and then divide the image into a grid cell with the size of $S \times S$ (7×7 default). If the center of an object falls into a grid cell, that grid cell predicts bounding boxes includes confidence probabilities for the objects.

The YOLO algorithm was upgraded to five versions (including the original version) since the time of its publication. In 2020, Bochkovskiy et. al [18] presented YOLOv4 with many of the most innovative ideas coming out of the computer vision research community for each part of the model architecture. After a few months, Glenn Jocher released YOLOv5. Unlike previous YOLO versions that picked the five best-fit anchor boxes for the COCO dataset and use them as default, YOLOv5 integrates the anchor box selection process by learning the best anchor boxes automatically for the training images and use them during training. Besides, YOLOv5 uses mosaic augmentation that combines four images into four tiles of random ratio, helping the models to detect small objects.

Faster R-CNN detector [19] usually comprises four major parts: 1) a feature extractor stage–usually using a CNN, 2) a region proposal (RPN) algorithm to predict bounding boxes of possible objects in the image with a confidence score, 3) a classification layer to predict which class this object belongs to, and 4) a regression layer to make the coordinates of the object bounding box more precise. The main idea of Faster R-CNN is using a convolutional network for RPN, instead of using a selective search algorithm to generate the region proposals that yield accelerating training time and improving feature representation.

2.3. Implementation

In our work, Firstly, the DBT images dataset was split into training and testing sets (patient-wise splitting). In the training phase, we apply data augmentation techniques mentioned in section 2.1 to produce two training sets (training set by data augmentation technique 1 and training set by data augmentation technique 2). For each of these training sets, we train each of the mentioned detectors individually. It is worth noting that for the YOLOv5 detector, we train four models with different sizes (YOLO-Small (S), YOLO-Medium (M), YOLO-Large (L), and YOLO-XLarge (XL)).

Secondly, in the testing phase, the trained models are used to predict bounding boxes for each DBT image in the test set. Each bounding box consists of a class that the object may belong to, a coordinates list $[x1, y1, x2, y2]$ and confidence score. The confidence represents scores reflect how confident the model is that the box contains an object, i.e. any Tumor in the box, P(Tumor). All predicted bounding boxes from a single DBT image are combined in a single bounding boxes list. Then, this list is passed to (NMS) algorithm to select the best bounding box out of a set of overlapping or duplicated boxes.

The selection criteria are based on the confidence probability threshold along with the intersection over union (IoU) overlap measure threshold.

To reduce the computational complexity, we resized the original DBT images to 640 × 640. In the training phase, we implement two data augmentation techniques to increase the number of training data. It worth noting that, in this work we used the pytorch implementation of the YOLOv5 [19] and Faster-RCNN [20]. We utilized the patch size 8, and the maximum number of training epochs was 50 for the YOLO model and 20 for the faster-RCNN model.

2.4. Evaluation Metrics

In this study, we use three of the most popular metrics used to evaluate object detection models [21]: the true positive rate (TPR), F1-score and PASCAL VOC mean average precision (mAP) to assess the performance of the lesion detection models. The true positive rate (TPR, also called sensitivity) is the probability that an actual positive will test positive and calculated By Equation 4.

$$TPR = \frac{TP}{TP+FN} \tag{4}$$

where TP, and FN is true positive and False Negative detections, respectivly. The F1 score is the harmonic mean of precision and recall, and calculated as follows

$$Precision = \frac{TP}{TP+FP}, \quad Recall = \frac{TP}{TP+FN} \tag{5}$$

$$F1-score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{6}$$

Where FP is the false positive detections, precision is the ratio of the number of true positives to the total number of positive predictions and recall is the same as TPR. The mean average precision (mAP) is precision averages across all recall values between 0 and 1 at various IoU thresholds. By interpolating across all points, mAP can be interpreted as the area under the curve of the precision-recall curve.

3. Experiments and discussion

3.1. Dataset

In our experiments, we used the DBTex challenge dataset [22]. It contains a total of 1000 breast tomosynthesis scans from 985 patients. Of 101 patients, only 224 DBT images have annotations (the rest images are not fully annotated). It should be noted that in our experiments we divided the DBT images dataset into training and testing sets (patient-wise): 71 patients for training and 30 patients for testing.

3.2. Experimental results

In the training phase, we used the two training sets produced from the two data augmentation techniques (Aug1 and Aug2). With Aug1, we produced a total of 1168 training DBT images, while with Aug2 we produced a total of 292 training DBT images.

Table 1 presents a quantitive comparison between the four YOLOv5 models (YOLO-S, YOLO-M, YOLO-L and YOLO-XL) and the faster R-CNN model trained on the training dataset produced by Aug1 in terms of TPR, F1-score, and mean average precision–mAP (IoU threshold = 5). As one can see, YOLO-S achieved the best lesion detection results when compared to the other YOLO models. However, faster R-CNN has more promising breast lesion detection results. It surpassed all YOLO models on all measures. These breast lesion detection results demonstrate that the faster R-CNN could be more suitable for DBT images, however, it has a higher computational complexity than YOLO models.

Table 1. The performance of the deep learning detection methods with Aug1.

	YOLOv5				Faster R-CNN
	S	M	L	XL	
TPR	34.8	31.8	24.2	22.7	50
F1-Score	48.5	45.7	36.1	35.7	54.1
mAP [iou=0.5]	31.8	34.1	26.2	26.7	45.1

Table 2. The performance of the deep learning detection methods with Aug2.

	YOLOv5				Faster R-CNN
	S	M	L	XL	
TPR	47	39.4	39.4	47	56.1
F1-Score	52.5	51.7	56.6	51.4	57.4
mAP [iou=0.5]	48.7	38.9	40.4	41.8	46.8

Table 2 compares the breast lesion detection results of the models when trained on the training dataset produced by Aug2 in terms of TPR, F1-score, and mAP. As shown, when comparing values from Table 2 to values from Table1, we can argue that training the deep learning detectors based on Aug2 can yields noticeable improvements on all TPR, F1-Score, and mAP metrics. With Aug2, the performance of the YOLO-S model was also increased by 17 points in terms of mAP. Besides, the TPR and F1-score of the faster RCNN were also advanced 6% and 3.3%, respectively.

Figure 3 presents the breast lesion detection results of the YOLO models and faster-RCNN. The red boxes stand for the ground truth while the green boxes indicate the detected lesions with confidence scores with IoU greater than 0.5. As shown, the breast lesion detections performed by the models trained with the Aug2 training dataset (second row) have higher confidence scores than models trained with the Aug1 training dataset (first row). In turn, the faster R-CNN predicts the most accurate bounding boxes ($IoU > 0.8$).

On the basis of the above analysis, we can conclude that to second data augmentation technique (Aug2) can significantly improve the breast lesion detection results for deep learning-based breast lesion detectors like YOLO models and faster R-CNN.

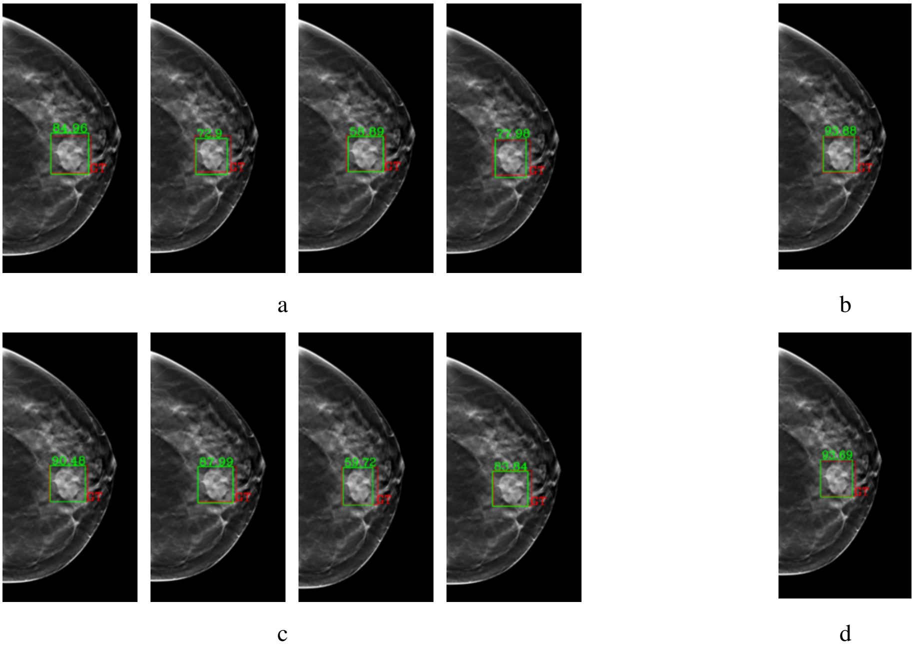


Figure 3. Examples of lesion detection using YOLOv5 and faster R-CNN. Note that the first row includes the results of Aug1 while the second row includes the results of Aug2. a and c show the detection results of YOLO-S, YOLO-M, YOLO-L and YOLO-XL from left to right, respectively. b and d show the detection results of Faster R-CNN.

All the experiments were performed using Pytorch framework using a 64-bit Ubuntu operating system with 3.6 GHz intel core i7 with 32GB of RAM and Nvidia GTX1080 with 8GB of video RAM.

4. Conclusions

In this paper, we investigated the strength of two data augmentation strategies (channel-replicate and channel-concatenation) while building five breast lesion detection models based on deep learning for digital breast tomosynthesis. We demonstrated that applying the channel-concatenation data augmentation strategy helps improve the detection accuracy of all deep learning models. With a publicly available digital breast tomosynthesis dataset, our experiments showed that the mAP score of YOLO models was increased by 17% approximately, while the F1-score of the faster-RCNN detector was improved by 3%. The future work will be focused on the development of a lesion detection approach based on the combination of robust deep learning-based detectors.

Acknowledgement

The Spanish Government partly supported this research through Project PID2019-105789RB-I00.

References

[1] H.D. Cheng, Juan Shan, Wen Ju, Yanhui Guo, and Ling Zhang. Automated breast cancer detection and classification using ultrasound images: A survey. *Pattern Recognition*, 43(1):299–317, 2010.

[2] Emine Devolli-Disha, Suzana Manxhuka-Kërliu, Halit Ymeri, and Arben Kutllovci. Comparative accuracy of mammography and ultrasound in women with breast symptoms according to age and breast density. *Bosnian journal of basic medical sciences*, 9(2):131–136, May 2009.

[3] James T Dobbins and Devon J Godfrey. Digital x-ray tomosynthesis: current state of the art and clinical potential. *Physics in Medicine and Biology*, 48(19):R65–R106, sep 2003.

[4] Mark A. Helvie. Digital mammography imaging: breast tomosynthesis and advanced applications. *Radiologic clinics of North America*, 48(5):917–929, Sep 2010.

[5] Julianne S. Greenberg, Marcia C. Javitt, Jason Katzen, Sara Michael, and Agnes E. Holland. Clinical performance metrics of 3d digital breast tomosynthesis compared with 2d digital mammography for breast cancer screening in community practice. *American Journal of Roentgenology*, 203(3):687–693, Sep 2014.

[6] Ming Fan, Yuanzhe Li, Shuo Zheng, Weijun Peng, Wei Tang, and Lihua Li. Computer-aided detection of mass in digital breast tomosynthesis using a faster region-based convolutional neural network. *Methods*, 166:103–111, 2019. Deep Learning in Bioinformatics.

[7] Muhammad Imran Razzak, Saeeda Naz, and Ahmad Zaib. *Deep Learning for Medical Image Processing: Overview, Challenges and the Future*, pages 323–350. Springer International Publishing, Cham, 2018.

[8] Dezső Ribli, Anna Horváth, Zsuzsa Unger, Péter Pollner, and István Csabai. Detecting and classifying lesions in mammograms with deep learning. *Scientific Reports*, 8(1):4165, Mar 2018.

[9] Jun Bai, Russell Posner, Tianyu Wang, Clifford Yang, and Sheida Nabavi. Applying deep learning in digital breast tomosynthesis for automatic breast cancer detection: A review. *Medical Image Analysis*, 71:102049, 2021.

[10] Ming Fan, Huizhong Zheng, Shuo Zheng, Chao You, Yajia Gu, Xin Gao, Weijun Peng, and Lihua Li. Mass detection and segmentation in digital breast tomosynthesis using 3d-mask region-based convolutional neural network: A comparative analysis. *Frontiers in molecular biosciences*, 7:599333–599333, Nov 2020.

[11] Xiaobo Lai, Weiji Yang, and Ruipeng Li. Dbt masses automatic segmentation using u-net neural networks. *Computational and Mathematical Methods in Medicine*, 2020:7156165, Jan 2020.

[12] William Lotter, Abdul Rahman Diab, Bryan Haslam, Jiye G. Kim, Giorgia Grisot, Eric Wu, Kevin Wu, Jorge Onieva Onieva, Yun Boyer, Jerrold L. Boxerman, Meiyun Wang, Mack Bandler, Gopal R. Vijayaraghavan, and A. Gregory Sorensen. Robust breast cancer detection in mammography and digital breast tomosynthesis using an annotation-efficient deep learning approach. *Nature Medicine*, 27(2):244–249, Feb 2021.

[13] Mateusz Buda, Ashirbani Saha, Ruth Walsh, Sujata Ghate, Nianyi Li, Albert Swiecicki, Joseph Y. Lo, and Maciej A. Mazurowski. Detection of masses and architectural distortions in digital breast tomosynthesis: a publicly available dataset of 5,060 patients and a deep learning model, 2021.

[14] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, 2016.

[15] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[16] Kitti Koonsanit, Saowapak Thongvigitmanee, Napapong Pongnapang, and Pairash Thajchayapong. Image enhancement on digital x-ray images using n-clahe. In *2017 10th Biomedical Engineering International Conference (BMEiCON)*, pages 1–4, 2017.

[17] Moi Hoon Yap, Manu Goyal, Fatima Osman, Robert Mart, Erika Denton, Arne Juette, and Reyer Zwiggelaar. Breast ultrasound region of interest detection and lesion localisation. *Artificial Intelligence in Medicine*, 107:101880, 2020.

[18] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv e-prints*, page arXiv:2004.10934, April 2020.

[19] Glenn Jocher. Yolov5 Pytorch Implementation. `https://github.com/ultralytics/yolov5`. Accessed: 2021-05-29.

[20] Faster-RCNN Pytorch Implementation. `https://pytorch.org/tutorials/intermediate/torchvision_tutorial.html`. Accessed: 2021-05-29.

[21] Rafael Padilla, Sergio L. Netto, and Eduardo A. B. da Silva. A survey on performance metrics for object-detection algorithms. In *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pages 237–242, 2020.

[22] SPIE-AAPM-NCI DAIR Digital Breast Tomosynthesis Lesion Detection Challenge. `https://www.aapm.org/GrandChallenge/DBTex/`. Accessed: 2021-05-23.

M. Villaret et al. (Eds.)

doi:10.3233/FAIA210151

Absolute Distance Prediction Based on Deep Learning Object Detection and Monocular Depth Estimation Models

Armin MASOUMIAN[a,1], David G. F. MAREI[a], Saddam ABDULWAHAB[a], Julián CRISTIANO[a], Domenec PUIG[a] and Hatem A. RASHWAN[a]
[a]*DEIM, Rovira i Virgili University, 43007 Tarragona, Spain*

Abstract. Determining the distance between the objects in a scene and the camera sensor from 2D images is feasible by estimating depth images using stereo cameras or 3D cameras. The outcome of depth estimation is relative distances that can be used to calculate absolute distances to be applicable in reality. However, distance estimation is very challenging using 2D monocular cameras. This paper presents a deep learning framework that consists of two deep networks for depth estimation and object detection using a single image. Firstly, objects in the scene are detected and localized using the You Only Look Once (YOLOv5) network. In parallel, the estimated depth image is computed using a deep autoencoder network to detect the relative distances. The proposed object detection based YOLO was trained using a supervised learning technique, in turn, the network of depth estimation was self-supervised training. The presented distance estimation framework was evaluated on real images of outdoor scenes. The achieved results show that the proposed framework is promising and it yields an accuracy of 96% with RMSE of 0.203 of the correct absolute distance.

Keywords. Deep learning, depth estimation, object detection, distance prediction

1. Introduction

For enabling fully autonomous driving and navigation, one of the main challenges is to achieve reliable and accurate obstacles detection. Many works have been proposed to cope with the problem of obstacles detection [1]. Object detection and distance prediction are effectively used in a variety of different fields such as industrial robots [2], robots for research [3], self-driving cars [4] and etc. Regarding object detection, to successfully navigate the environment, the moving systemmust have knowledge about the objects in its immediate vicinity. Among many sensors available for object detection (e.g. LIDAR sensors) we are primarily interested in a camera-based vision for indoor/outdoor navigation. Thus, object recognition based cameras refers to a collection of related tasks for identifying objects in digital photographs.

With the progress of deep learning networks (e.g., Convolutional Neural Networks (CNN)), many accurate methods for object recognition have been developed. For instance, region-Based CNN, or R-CNNs [5], are a family of techniques for addressing object localization and recognition tasks, designed for model performance. In turn, You

[1] Corresponding Author: E-mail: masoumian.armin@gmail.com

Only Look Once, or YOLO [6], is a second family of techniques for object recognition designed for a fast responseand real-time applications.

Region-based detectors include two stages. Firstly, the model suggests a set of regions of interests (ROIs) by a regional proposal network. Since the potential bounding box candidates can be infinite, the proposed regions are sparse. Secondly, the region candidates are then processed by a classifier. In turn, the one-stage family skips the region proposal stage and it directly runs the detection over a dense sampling of possible locations. This yields that the one-stage detectors are faster and simpler, but might potentially reduce the performance a bit. Since YOLO has the advantage of being much faster than other networks in the one-stage family. Besides it achieved comparable results to the state of the art and still maintains accuracy. The predictions depend on the global context of the input image. Consequently, our proposed framework will be based on the YOLO architecture as a baseline.

Regarding distance prediction, it is important to estimate depth maps from the input images. For depth estimation, most computer-vision systems depend on stereo vision by following several time-consuming stages, such as unipolar geometry, rectification and matching. Alternatively, when stereo vision is not useful or applicable, LIDAR cameras can be used for many applications for mobile robots. However, LIDAR sensors are very costly, and most depth cameras have serious limitations in real environments, such as the synchronization of the optical and imaging elements [7]. With the deep learning spread, many works have been proposed for monocular depth estimation that is the task of estimating scene depth using a single image. The appearance of objects significantly changes with their pose. Estimating a depth map from a 2D image is an important step in order to determine the 3D pose of the objects present in a scene. Monocular depth estimation based on deep learning methods can be performed by supervised [8] or unsupervised [9] learning techniques. Supervised methods perform better accuracy, however the depth maps of images is needed for training which is difficult to get in real scenarios. On the other hand, unsupervised methods do not require original depth maps, thus, the performance is degraded a bit.

Thus, in this paper, we propose a new framework to predict the absolute distance of each object in 2D images captured from a monocular camera, based on estimating depth images using self-supervised deep learning and supervised deep learning object detection. The contributions of this paper are:

- Developing a deep object detection model based on two-stage YOLOv5 architecture. A lightweight model that can be easily deployed on embedded systems and devices with a limited memory and CPU.
- Developing an unsupervised depth and pose estimation deep learning model based on an autoencoder network.
- Integrating the two models in a framework for absolute distance estimation of obstacles. Integrating the two models will not affect the overall efficiency of the proposed models, because the two models are structurally independent, and the whole framework is executed by multi processes, meaning that each model has a separate process responsible for it.

Section 2 introduces previous background review, section 3 explains our proposed methodology and section 4 describes our performed experiments.

2. Background Review

In this section, we summarize the state-of-the-art for both systems: depth estimation and object detection.

2.1. Object Detection

Object detection is a computer vision technique that allows a system to locate and identify an object in an image or video and detecting a bounding box around each one of them. It is one of the most challenging issues in the field of computer vision as the object detection model is trained to identify objects within a dataset and it cannot identify an object that is not labeled during the training and this is considered as one of the limitations. However, trained object detection models can always be retrained again to obtain new knowledge about new objects. Object detection techniques are used in applications like self-driving cars, video surveillance or crowd counting. There are some popular object detection algorithms like YOLO [6], R-CNN [5] and MobileNet [10]. In this paper the YOLO Algorithm has been chosen for object detection. As it is considered as the state-of-the-art right now and it produced the needed results in the testing phase.

The YOLO object detection model has several different versions through the years. The YOLOv1 paper was published in 2015 and later the subsequent versions were published the next years until it reached YOLOv5 in 2020 and it is considered as the state-of-the-art due to its good performance and efficiency and constantly being improved.

2.2. Depth Prediction

Depth and ego motion estimation is one of the fundamental challenges in computer vision. Two different methods exist to achieve these goals, supervised deep learning and unsupervised deep learning models.

2.2.1. Supervised Depth Estimation

Predicting a depth from a single image is an innately difficult task as the same image can project multiple conceivable depths. To prevent this, predicted depth needed to have some relationship with color image. There are various approaches such as end to end [11], sense sampling of non- parametric [12], optical flow [13], transfer learning [14] and combining local predictions [15], have been done. In supervised methods, the original depth maps of images are used to train alongside color images. This helps the system to learn better and therefore the results of supervised methods usually have better performance than unsupervised methods. However in reality it is difficult to construct the depth maps in real time, and to do that, the use of a stereo camera or 3D LIDAR is necessary.

2.2.2. Unsupervised Depth Estimation

To avoid the aforementioned problems, unsupervised methods have been used for training the systems, only original images and pre trained models such as DenseNet

[16], ResNet [17], and ImageNet [18] are needed. Regarding unsupervised methods, various approaches for depth estimation have been proposed, such as generative adversarial networks [19], temporal information [20] and separate pose networks [21].

3. Method

In this work, as shown in Figure 1, we used two parallel deep networks: one for object detection and the other for depth estimation. The predicted depth is extracted from DepthNet. In turn with YOLOv5, the objects inside the image is localized and classified. Furthermore, the localization of each object defined by bounded boxes is detected on the estimated depth image. Finally, the relevant distance of an object is calculated by the median estimated distance of all pixels inside the defined bounded box.

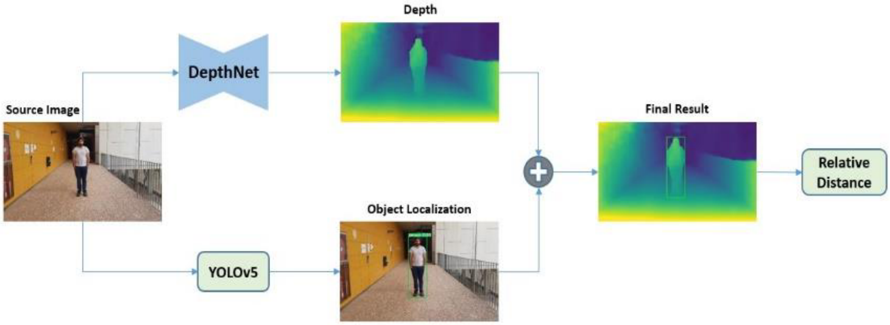


Figure 1. An illustration of the overall framework

Regarding DepthNet, it has two networks: one for estimating depth images and the other for estimating the image pose. Both networks are based on autoencoder networks that consist of two serial networks: encoders and decoders. For encoder of depth and pose networks, the ResNet pre-trained weight has been used for extracting the features and representing the input images. The ResNet-50 has been used as a backbone network for our depth prediction, while ResNet-18 has been used for pose estimation. In both networks, the first step before entering the first layer of the ResNet is a block — called here Conv1 — consisting on a convolution + batch normalization + max pooling operation. Then, four blocks of the ResNet are used. Regarding the decoder, each layer consists of a deconvolutional and upsampling, as shown in Figure 2. The last layer of decoder is the estimated depth map. Besides, we used our initial work based on Graph Convolutional Networks (GCN), which is one of the most powerful neural network architecture. GCN can properly find the similarity of the pixels and make a graph connection between them [22]. The proposed GCN model learns the features by inspecting neighboring nodes. The GCN network is used in the decoder network to construct accurate depth images in multi-scale as shown in Figure 2.

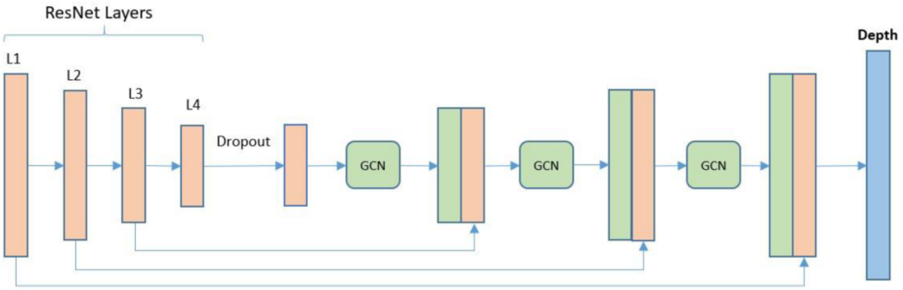


Figure 2. Overview of DepthNet network architecture

Regarding object detection, YOLOv5 is designed to create features from input images and later on feed these features through a prediction system to draw boxes around objects and predict their classes. This architecture consists of three main parts: Backbone, Neck and Head. YOLOv5 employed state-of-the-art network EfficientNet [23] as its backbone, making that the model has sufficientability to learn the complex features of input images. YOLOv5 applied an improved PANet[24], named bi-directional feature pyramid network (Bi-FPN) as its neck, to allow easy and fast multi-scale feature fusion. Bi-FPN introduces learnable weights, enabling the network to learn the importance of different input features, and repeatedly applies top-down and bottom-up multi-scale feature fusion. Thirdly, YOLOv5 integrates a compound scaling method that uniformly scales the resolution, depth, and width for all backbone, feature network, and box/class prediction networks at the same time, which ensures maximum accuracy and efficiency under the limited computing resources. Figure 3 demonstrates the architecture of YOLOv5 that we used to detect objects.

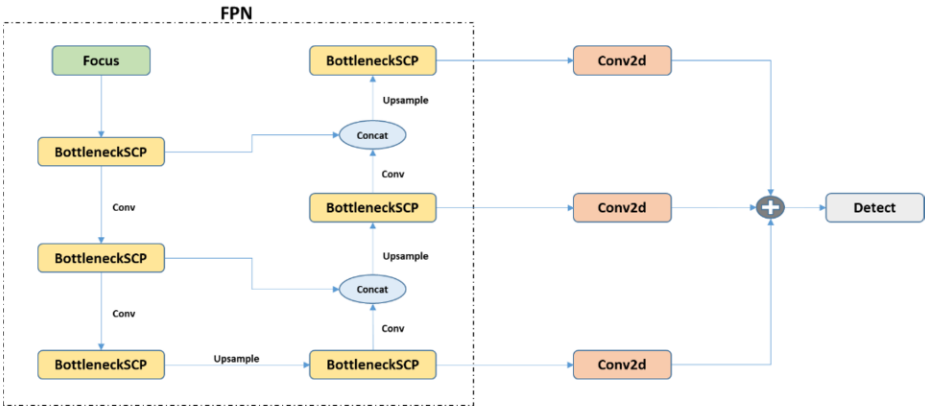


Figure 3. Overview of YOLOv5 network architecture.

3.1. Absolute Distance Prediction

After training our depth predicted model with a KITTI dataset [25], the model was tested with our private own dataset, explained in detail in section 4, to estimate the depth of each image in the testing set. At the same time, with the YOLOv5 model, objects in the image have been detected and the location of the bounding box was determined. Based on the bounding box coordinates, the exact box was localized on the corresponding predicted depth image. Indeed, the DepthNet estimates the disparity maps that represent the relation between the motion between the pixels of the input image and the ones of the target image (i.e., could be a consequence image). So, we transform the disparity maps to depth maps [26]. In the DepthNet network, minimum depth and maximum depth have been set as 0 to 100 meters same as the KITTI dataset.

Afterwards, the median value of the estimated distances of all pixels inside the bounding box of an object in a depth imageis computed. This estimated distance can be named the relative distance of an object (REV).

However to convert the REV distance to absolute distance (ABS), the real distance of objects in images are needed. Thenceforth, the relation between the absolute distance and relative distance have to be calculated. In most works of ABS estimation, ABS of an object depends on the type and shape of objects, as well as the image size and focal length of the sensor [27]. However, in this work, we need to avoid depending on this type of information and we will try to calibrate our method to work for different unknown objects.

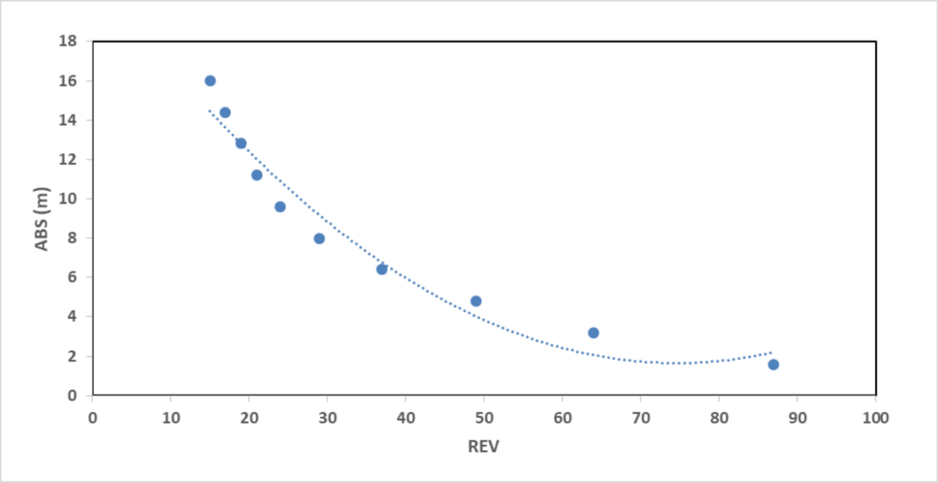


Figure 4. Perform the relation between ABS and REV

Consequently, based on the Taha and Jizat technique [28], the ABS distance can be calculated based on a mathematical quadratic function:

$$Y = (c_0 + c_1X + c_2X^2) \times h \tag{1}$$

Where c_0, c_1, c_2 coefficients can be obtained using least square equations, h is the camera height, and X is the relevant distance from the object to the beginning of the camera's field of view. Based on this hypothesis, curve fitting and least-squared optimization is applied to find the approximated value of the four unknown coefficients. The solution has the best fit to a series of data points (i.e., we used 10 images with different objects and distances)to find the relation between ABS and REV distances, as shown in Figure 4:

$$Y = 0.0036X^2 - 0.5373X + 21.714 \tag{2}$$

4. Experiments

Implementation Details: We implemented a code of depth estimation in Pytorch. The depth estimation model trained for 20 epochs, a batch size of 10, a learning rate of 0.0001 and the Adam optimizer. The training process took 5 days using a single GPU of GTX 1080 TI. For object detection with YOLOv5, we used also the Pytorch library with 80 different classes. Regarding the pre-trained checkpoints, YOLOv5 with a light version (YOLOv5s) has been chosen due to its lower computational cost.

4.1. Datasets

KITTIdataset is one of the famous computer vision dataset for depth and pose estimation. The dataset contains 200 videos of street scenes in day light captured by RGB cameras and the depth maps captured by Velodyne laser scanner. We used synchronized single images from a monocular camera and Eigen split [29] with 39810 images for training, 4424 for validation and 697 images for testing. The [21] preprocessing method has been used for removing static frames. The resolution input size of the images is 320 pixels × 1024 pixels.

Cocodataset [30] is one of the large-scale object detection, segmentation and captioning dataset. The dataset contains 80 different object classes with a total of 2.5 million labeled instances in 328k images. We used the original split dataset with 165482 images for training, 81208 for validation and 81434 images for testing. Theimages sizes are 640 pixels × 480 pixels.

From the predicted depths, the relative distance of each pixel can be extracted. However, having an absolute distance is necessary to first find the relation function and second evaluate the results. Therefore, we prepared our **own private dataset** that contains 100 images (with a resolution of 777 pixels × 1350 pixels) with a hand-held camera. Monocular RGB camera was mounted on a static stand and the absolute distance of each object was manually measured from the camera. The absolute distance from the camera and objects have manually been defined of all objects in scenes. During the collection of the dataset, we imitated potential static obstacles on the front of the camera. These obstacles were located in different distances from the camera test-stand.

4.2. Evaluation

During the test procedure, the performance of the proposed method was checked by using 100 images with different objects such as person, car, chair, etc. (i.e., remember we can detect 80 classes of objects as COCO used).

We used two standard evaluation measures to assess the proposed framework: Accuracy and Root Mean Square Error (RMSE). The Accuracy was used to estimate errors under a given threshold, serving as an indication of how often our estimation is correct. The threshold accuracy measure from [31] is essentially the expectation that the absolute distance value error of a given object in a scene is lower than a threshold T (i.e., in this work T is set by 0.2 m).

Figure 5 demonstrates the qualitative results of the proposed framework and it shows some examples of our own private dataset including the original images, estimated depth images, the results object localization using YOLOv5 and the relevant depth estimated by DepthNet. In turn, Table 1 represents the measurement of absolute distance vs. predicted distance. It is obvious from Figure 5, object detection based YOLOv5 is reliable in spite of the fact that YOLO classifier was used in its original form trained with COCO dataset, without fine-tuning with the images from own private dataset. Besides, it is obvious from Table 1 that achieved absolute distance estimation is satisfactory in spite of the fact that our own private dataset did not contain object boxes (object localization) from the captured scenes.

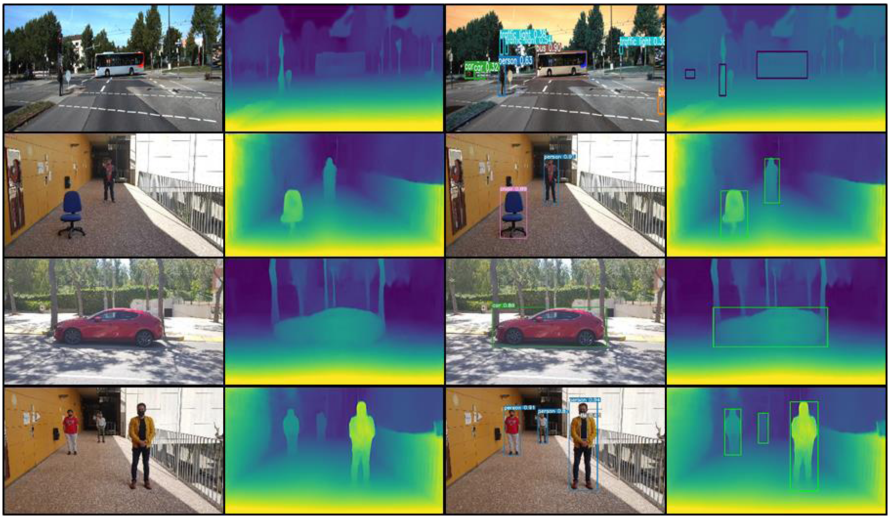


Figure 5. Visual process of whole network

Table 1. Estimated distance vs. absolute distance. Note the objects are counted in the tested images from left to right.

Figure 5	Object	Absolute distance (m)	Predicted distance (m)	Error (m)
Row 1	Car	53.9	53.21	0.69
	Person	21.5	21.35	0.15
	Bus	48.7	48.13	0.57
Row 2	Chair	3.5	3.45	0.05
	Person	8.0	8.09	0.09
Row 3	Car	10.1	9.83	0.27
Row 4	Person 1	8.0	8.13	0.13
	Person 2	12.0	11.69	0.31
	Person 3	4.0	3.88	0.12

As it is shown in Table 1, the farther away the objects are, the more error we will get. The proposed framework achieved an accuracy and average RMSE of 96% and 0.203 (m), respectively. It is obvious that the predicted distance is satisfactory despite the fact that our private dataset was not part of the training dataset for DepthNet and YOLOv5.

In contrast to the DispNet method [27], which detected only the objects on railways, the presented framework recognized different objects in the scene recorded by the own private dataset. As obvious, for big objects (e.g., cars), the YOLO network can easily detect them even with a large distance. However for small objects (e.g., chair or person), YOLO is sometimes not able to detect them from a large distance that degrade the performance of the proposed framework. Thus, in the future work, the YOLO will be updated to work with tiny objects.

5. Conclusion

In this paper, we proposed a reliable deep framework for estimating absolute distances of objects in real scenes. Our methods consist of 2 parallel networks; the first one is used to predict the depth values of images using a 2D monocular camera based on an unsupervised autoencoder network, and the second one detects the objects and extracts its localization box within the scene. Furthermore, the absolute distance of an object can be computed from relative distance by calibrating our framework with real images. Ongoing work, the absolute distance estimation will be achieved based on a learnable network for generalizing the framework for different objects and shapes and thus to improve the accuracy of distance estimation in own private dataset. Future work aims at developing an intelligent assistant system for aiding visual impaired people based on the proposed framework.

Acknowledgments

This research has been possible with the support of the Secretariad' Universitatsi Recercadel Departamentd' Empresai Coneixement de la Generalitat de Catalunya (2020 FISDU 00405). We are thankfully acknowledging the use of the University of Rovira I Virgili (URV) facilities to carry out this work.

References

[1] Badue C, Guidolini R, Carneiro R V, Azevedo P, Cardoso VB, Jesus LFR, et al. Self - Driving Cars A Survey - 2017(2).pdf. 2017;(January).

[2] Andhare P, Rawat S. Pick and place industrial robot controller with computer vision. In: Proceedings - 2nd International Conference on Computing, Communication, Control and Automation, ICCUBEA 2016. Institute of Electrical and Electronics Engineers Inc.; 2017.

[3] Masoumian A, Montazer MC, Valls DP, Kazemi P, Rashwan HA. Using the Feedback of Dynamic Active-Pixel Vision Sensor (Davis) to Prevent Slip in Real Time. In: 2020 6th International Conference on Mechatronics and Robotics Engineering, ICMRE 2020. Institute of Electrical and Electronics Engineers Inc.; 2020. p. 63–7.

[4] Agarwal N, Chiang CW, Sharma A. A Study on Computer Vision Techniques for Self-driving Cars. In: Lecture Notes in Electrical Engineering [Internet]. Springer Verlag; 2019 [cited 2021 May 7]. p. 629–34. Available from: https://link.springer.com/chapter/10.1007/978-981-13-3648-5_76

[5] Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Trans Pattern Anal Mach Intell [Internet]. 2017 Jun 1 [cited 2021 May 7];39(6):1137–49. Available from: http://image-net.org/challenges/LSVRC/2015/results

[6] Redmon J, Divvala S, Girshick R, Farhadi A. You Only Look Once: Unified, Real-Time Object Detection. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit [Internet]. 2015 Jun 8 [cited 2021 May 7];2016-December:779–88. Available from: http://arxiv.org/abs/1506.02640

[7] Olanrewaju HG, Popoola WO. Effect of Synchronization Error on Optical Spatial Modulation. 2017;65(12):5362–74.

[8] Abdulwahab S, Rashwan HA, Garcia MA, Jabreel M, Chambon S, Puig D. Adversarial Learning for Depth and Viewpoint Estimation from a Single Image. IEEE Trans Circuits Syst Video Technol. 2020;30(9):2947–58.

[9] Shu C, Yu K, Duan Z, Yang K. Feature-metric Loss for Self-supervised Learning of Depth and Egomotion. Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics) [Internet]. 2020 Jul 21 [cited 2021 May 6];12364 LNCS:572–88. Available from: http://arxiv.org/abs/2007.10603

[10] Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. arXiv [Internet]. 2017 Apr 16 [cited 2021 May 20]; Available from: http://arxiv.org/abs/1704.04861

[11] Laina I, Rupprecht C, Belagiannis V, Tombari F, Navab N. Deeper Depth Prediction with Fully Convolutional Residual Networks. Proc - 2016 4th Int Conf 3D Vision, 3DV 2016 [Internet]. 2016 Jun 1 [cited 2021 May 7];239–48. Available from: http://arxiv.org/abs/1606.00373

[12] Karsch K, Liu C, Kang SB. Depth extraction from video using non-parametric sampling. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) [Internet]. Springer, Berlin, Heidelberg; 2012 [cited 2021 May 7]. p. 775–88. Available from: http://cs.nyu.edu/

[13] Ilg E, Mayer N, Saikia T, Keuper M, Dosovitskiy A, Brox T. FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks. Proc - 30th IEEE Conf Comput Vis Pattern Recognition, CVPR 2017 [Internet]. 2016 Dec 6 [cited 2021 May 7];2017-January:1647–55. Available from: http://arxiv.org/abs/1612.01925

[14] Alhashim I, Wonka P. High Quality Monocular Depth Estimation via Transfer Learning. arXiv [Internet]. 2018 Dec 31 [cited 2021 May 7]; Available from: http://arxiv.org/abs/1812.11941

[15] Saxena A, Sun M, Ng AY. Make3D: Learning 3D scene structure from a single still image. IEEE Trans Pattern Anal Mach Intell. 2009;31(5):824–40.

[16] Huang G, Liu Z, van der Maaten L, Weinberger KQ. Densely Connected Convolutional Networks. Proc - 30th IEEE Conf Comput Vis Pattern Recognition, CVPR 2017 [Internet]. 2016 Aug 24 [cited 2021 May 7];2017-January:2261–9. Available from: http://arxiv.org/abs/1608.06993

[17] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition [Internet]. IEEE Computer Society; 2016 [cited 2021 May 7]. p. 770–8. Available from: http://image-net.org/challenges/LSVRC/2015/

[18] Jia Deng, Wei Dong, Socher R, Li-Jia Li, Kai Li, Li Fei-Fei. ImageNet: A large-scale hierarchical image database. 2009;248–55.

[19] Pilzer A, Xu D, Puscas MM, Ricci E, Sebe N. Unsupervised Adversarial Depth Estimation using Cycled Generative Networks. Proc - 2018 Int Conf 3D Vision, 3DV 2018 [Internet]. 2018 Jul 28 [cited 2021 May 7];587–95. Available from: http://arxiv.org/abs/1807.10915

[20] Madhu Babu V, Das K, Majumdar A, Kumar S. UnDEMoN: Unsupervised Deep Network for Depth and Ego-Motion Estimation. In: IEEE International Conference on Intelligent Robots and Systems. Institute of Electrical and Electronics Engineers Inc.; 2018. p. 1082–8.

[21] Zhou T, Brown M, Snavely N, Lowe DG. Unsupervised Learning of Depth and Ego-Motion from Video. Proc - 30th IEEE Conf Comput Vis Pattern Recognition, CVPR 2017 [Internet]. 2017 Apr 25 [cited 2021 May 7];2017-January:6612–21. Available from: http://arxiv.org/abs/1704.07813

[22] Kipf TN, Welling M. Semi-Supervised Classification with Graph Convolutional Networks. 5th Int Conf Learn Represent ICLR 2017 - Conf Track Proc [Internet]. 2016 Sep 9 [cited 2021 May 7]; Available from: http://arxiv.org/abs/1609.02907

[23] Tan M, Le Q V. EfficientNet: Rethinking model scaling for convolutional neural networks. 36th Int Conf Mach Learn ICML 2019. 2019;2019-June:10691–700.

[24] Liu S, Qi L, Qin H, Shi J, Jia J. Path Aggregation Network for Instance Segmentation. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit. 2018;8759–68.

[25] Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? the KITTI vision benchmark suite. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2012. p. 3354–61.

[26] Uhrig J, Schneider N, Schneider L, Franke U, Brox T, Geiger A. Sparsity Invariant CNNs. Proc - 2017 Int Conf 3D Vision, 3DV 2017. 2018;11–20.

[27] Haseeb MA, Guan J, Ristić D, Gräser A. DisNet : A novel method for distance estimation from monocular camera. 10th Planning, Percept Navig Intell Veh. 2018;

[28] Taha Z, Jizat JAM. A comparison of two approaches for collision avoidance of an automated guided vehicle using monocular vision. Appl Mech Mater. 2012;145(January):547–51.

[29] Eigen D, Fergus R. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. Proc IEEE Int Conf Comput Vis. 2015;2015 Inter:2650–8.

[30] Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, et al. Microsoft COCO: Common objects in context. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) [Internet]. Springer Verlag; 2014 [cited 2021 May 7]. p. 740–55. Available from: https://arxiv.org/abs/1405.0312v3

[31] Liu F, Shen C, Lin G. Deep convolutional neural fields for depth estimation from a single image. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit. 2015;07-12-June(November 2014):5162–70.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210152

Benchmarking Deep Learning Models on Point Cloud Segmentation

Thanasis ZOUMPEKAS [a,1], Guillem MOLINA [a], Maria SALAMÓ [a] and Anna PUIG [a]
[a] *Department of Mathematics and Computer Science, University of Barcelona, Barcelona, Spain*

Abstract. Point clouds are currently used for a variety of applications, such as detection tasks in medical and geological domains. Intelligent analysis of point clouds is considered a highly computationally demanding and challenging task, especially the segmentation task among the points. Although numerous deep learning models have recently been proposed to segment point cloud data, there is no clear instruction of which exactly neural network to utilize and then incorporate into a system dealing with point cloud segmentation analysis. Besides, the majority of the developed models emphasize more on the accuracy rather than the efficiency, in order to achieve great results. Consequently, the training, validation and testing phases of the models require a great number of processing hours and a huge amount of memory. These high computational requirements are commonly difficult to deal with for many users. In this article, we analyse five state-of-the-art deep learning models for part segmentation task and give meaningful insights into the utilization of each one. We advance guidelines based on different properties, considering both learning-related metrics, such as accuracy, and system-related metrics, such as run time and memory footprint. We further propose and analyse generalized performance metrics, which facilitate the model evaluation phase in segmentation tasks allowing users to select the most appropriate approach for their context in terms of accuracy and efficiency.

Keywords. Deep Learning, Neural Networks, Point Clouds, Segmentation, Analysis, Benchmark

1. Introduction

Nowadays, point cloud data is utilized as input for a great assortment of applications, including the creation of 3D models for manufactured parts, detection and identification procedures in medical, geological and autonomous-driving domains and engineering simulations [1]. A point cloud is a set of points in 3D space, characterized by the x, y, and z coordinates, that can be mainly acquired by 3D sensors, such as LIDARs and RGB-D cameras. These devices are able to capture with detail surface and geometrical properties of objects [2]. Indeed, point cloud data of a 3D object may contain millions of points with highly detailed information, but they also contain scattered, disjointed information and in most cases with a lot of noise [1].

[1] Corresponding Author: Department of Mathematics and Computer Science, University of Barcelona, Barcelona, Spain; E-mail: thanasis.zoumpekas@ub.edu

In fact, the rapid advancements of the 3D sensing technology in recent times are increasing the demand for 3D point cloud processing techniques in order to extract implicit information. In this sense, deep learning approaches are the most frequently used techniques providing intelligent solutions in a wide range of applications. However, the intelligent analysis of such huge data is a highly computationally demanding and complex task, especially the segmentation task among the points [3]. The segmentation process is the categorization of the points of an object into different parts and groups. It is based on the notion that the points belonging to a specific area of an object have the same properties. The segmentation of scattered and huge data such as point clouds is a challenging process and usually segmentation models mainly focus on the improvement of accuracy [4] rather than efficiency.

Numerous intelligent approaches achieve great accuracy in segmenting parts of objects and whole scenes of point cloud data [5,6]. While some of them include measurements of the computational power per second (i.e. FLOPS), such as [6], or investigate memory costs of operations[7], there is still a big open question on the efficiency of the accurate models. It is considered highly important to include all feasible metrics for a proposed model, as pointed out by [8], but to the best of our knowledge, at the time of this writing, there is no specific performance evaluation approach that evaluates the point cloud data segmentation accuracy of a model concerning the time and memory allocation needed to achieve the best accuracy score. In this article, we address this issue and experiment with five of the most accurate Neural Networks[2] for segmenting parts of objects represented with point cloud data. Specifically, we analyse PointNet [9], PointNet++ [10], Kernel Point Convolution (KPConv) architecture [5], Position Pooling (PosPool) architecture [7] and Relation-Shape Convolutional (RSConv) neural network [6].

Thus, our paper makes three contributions: **(i)** We analyse the performance of the used models in relation to time and average memory allocation, apart from accuracy, **(ii)** We propose and formulate novel performance metrics taking into account both *learning-related*, such as *mIoU*, and *system-related* metrics, such as run time and memory footprint, **(iii)** We analyse in-depth these metrics to define clear and meaningful insights on the accuracy and efficiency of the models and the trade-off between them.

2. Related Work

Point cloud segmentation using deep learning models is an emerging study field and have recently caught the attention of a vast majority of researchers, mainly due to the presence of unique challenges, such as the lack of structure and the high dimensionality of the data [3]. Apart from that, numerous review studies present multiple deep learning models handling point clouds with a great accuracy [3,1,11]. The groundbreaking neural network dealing with such tasks was PointNet [9] and after its release, a great extent of models came out improving the segmentation and classification accuracy of it [11].

In the original study of PointNet, Qi et al. [9] include a short time and complexity analysis of it. However, their analysis do not take into consideration multiple comparison dimensions among the other published neural networks. More recently, Liu et al. [7], compare their proposed model to other studies showing a benchmark that is mostly fo-

[2] https://paperswithcode.com/sota/3d-part-segmentation-on-shapenet-part

cused on accuracy and endogenous parameters of the neural networks, for example, width and depth of the architectures. In addition, the majority of the new and state-of-the-art models, such as [12,5,10], focus primarily on the improvement of the segmentation accuracy, providing basic or no information regarding efficiency.

There are also approaches, that consider other metrics apart from accuracy of the model. Coleman et al. [13] present an evaluation approach of deep learning models taking into account *system-related* and *learning-related* performance metrics, such as training time and accuracy respectively. They provide a benchmark that compares the training time needed to achieve the state-of-the-art accuracy, as well as, the the time it takes to infer having the best accuracy. Additionally, an extensive research of time-to-accuracy performance analysis is presented in [14]. However, these studies focus on image classification and segmentation tasks, involving imagery data, i.e. 2D data but not point clouds.

On the other hand, Garcia-Garcia et al. [8] present an interesting study on deep learning techniques with an application to segmentation task. They put on foreground not only the accuracy but also the efficiency of a wide range of methods involving not just 2D image data but even 3D point cloud data. They highlight that there is a need for performance evaluation of segmentation models to be considered useful and valid in practice. Even though, they claim that the most important metrics for the evaluation of deep learning segmentation models should be the execution time, the memory footprint and the accuracy, stating that the trade-off among them should be parameterized depending on the system's or analysis' objective, their analysis provide only accuracy-related results.

We consider there is a gap in benchmarking in-depth segmentation models on point cloud data, taking into consideration dimensions such as execution time, memory footprint and accuracy, that allow users to validate and select the appropriate model for their specific application or context.

3. Proposal

We propose a performance benchmark for deep learning algorithms for the task of point cloud part segmentation. Our main inspiration is derived from [13,8] and the lack of studies highlighting which is the best Neural Network in terms of accuracy and efficiency for the part segmentation task. Please note, that by the term accuracy in the segmentation process, we denote the performance in the metric of point *Intersection over Union* (IoU), which is explained onwards. Also, the term efficiency describes the least amount of resources of a method in run time and memory footprints. Therefore, we experiment with a well-known point cloud dataset, and we analyse five of the most accurate deep learning models providing not only segmentation accuracy, i.e learning-related metrics, but also system-related metrics, such as run time and memory footprints.

We acknowledge that accuracy is necessary in any intelligent system. However, other performance dimensions should also be considered and evaluated. Thus, we propose novel performance metrics aiming to balance the decision border between accuracy and efficiency, taking into consideration both learning-related and system-related metrics.

Considering learning-related metrics, the point *Intersection over Union* (IoU) performance metric is the most typical metric to evaluate 3D point cloud part segmentation models. IoU is simply defined as $IoU = \frac{A \cap B}{A \cup B}$, where A is the area of points of ground-truth point cloud and B of the predicted point cloud. It appears in the majority of the research studies working in the aforementioned field, such as [9,10,5,7,6]. Specifically, two variants of the IoU metric are the most used ones, the mean Intersection over Union ($mIoU$) obtained by averaging across all Classes ($CmIoU$) and all Instances ($ImIoU$). For clarification, the $ImIoU$ is the average of the $mIoU$ across all point clouds, i.e. all learning instances. The $CmIoU$ metric is the average of the $mIoU$ across all point clouds belonging to specific classes and then averaged again on the number of classes. The details of the aforementioned metrics are also described in [6].

Aside from the learning-related performance metrics, we further measure the system-related metrics such as run time and average amount of memory allocated by the Graphics Processing Unit (GPU) of the whole learning procedure of each utilized neural network.

3.1. Proposed Metric

Specifically, we propose the F_{CmIoU}, F_{ImIoU} and $F_{general}$ and we analytically explain the formulation of them below, in equations (1), (2) and (3) respectively. We formulate the F_{CmIoU} (Eq. 1) that provides a per class segmentation accuracy ($CmIoU$) as well as the efficiency of the model involving the total run time (t_{total}), which is the total time spent to finish the whole learning process of training, validation and testing, and the average percentage of GPU memory allocation (GPU_{mem}) that is used for computations in the whole learning process. In a similar way, we formulate the F_{ImIoU} (Eq. 2), which aims to balance the decision border in model selection among $ImIoU$, t_{total} and GPU_{mem}. Finally, in an attempt to give a more generalized metric facilitating further the model selection procedure, considering both general accuracy and efficiency of the models, we propose the $F_{general}$ (Eq. 3), which is the arithmetic mean between the F_{CmIoU} and F_{ImIoU}. For comparison purposes across the different metrics, we normalize all the values per metric in the range $[0,1]$.

In F_{CmIoU} (Eq. 1), we denote β the coefficient that balances the decision border among $CmIoU$, t_{total} and GPU_{mem} metrics. Higher β values portray more focus on the learning-related metric of $CmIoU$. In the notion of system-related metrics, i.e time and memory, the best model is considered to be the one having the lowest time duration and with the smallest memory footprint. Thus, in an attempt to highlight this, we formulate the time-related component in the equation as $(1-t_{total})$ and the memory-related component as $(1-GPU_{mem})$. Likewise, in F_{ImIoU} (Eq. 2) we denote α the balance coefficient among $ImIoU$, t_{total} and GPU_{mem}. Note that both β and α coefficients take values in the range of $[0,1]$ and the sum of the weight factors in the equations is equal to 1, i.e. $\beta+\frac{(1-\beta)}{2}+\frac{(1-\beta)}{2}=1$ and $\alpha+\frac{(1-\alpha)}{2}+\frac{(1-\alpha)}{2}=1$. All the proposed metrics, i.e. F_{CmIoU}, F_{ImIoU} and $F_{general}$, take values in the range $[0,1]$.

$$F_{CmIoU}=\beta*CmIoU+\frac{(1-\beta)}{2}*(1-t_{total})+\frac{(1-\beta)}{2}*(1-GPU_{mem}) \tag{1}$$

$$F_{ImIoU}=\alpha*ImIoU+\frac{(1-\alpha)}{2}*(1-t_{total})+\frac{(1-\alpha)}{2}*(1-GPU_{mem}) \tag{2}$$

$$F_{general} = \frac{F_{CmIoU} + F_{ImIoU}}{2} \tag{3}$$

Note that the proposed metrics can be used stand-alone (Eqs. 1 and 2) or combined in a unique metric (Eq. 3) for more generalized performance evaluation, depending on the objective of the application.

4. Evaluation

This section presents the segmentation models, the data used and the benchmark protocol in our evaluation process. Finally, we show and analyse our results.

4.1. Models

We investigate five of the most accurate neural networks dealing with 3D point cloud part segmentation. Following, we briefly describe them.

PointNet [9] is considered one of the fundamental and pioneering deep learning architectures on point cloud data, which handles both classification and segmentation tasks using point-wise operations. It focuses on the global structure of a point cloud while being symmetrical or invariant by the input order of point clouds. Additionally, it features point-wise robustness against noisy elements, perturbation or missing data. However, the point-wise feature extraction operations of PointNet do not take into consideration the topology of the points, i.e. neighbourhoods of points.

PointNet++ [10] is the successor of the PointNet architecture that is capable to focus on both the global and local structure of a point cloud object. Moreover, it also considers the geometrical properties and is robust to size variance and point density variance per point and per neighbourhood of points. One strong point of PointNet++ is that it is adequate to deal with surfaces of point clouds.

KPConv [5] approach emphasizes on local aggregation computations among the points. Specifically, it uses kernel operations, which are defined by a set of points distributed in a sphere. The convolutions are based on a defined metric system and are dependent on the dimensions of the input data. Thus, they need to be adjusted to adequate sizes for each problem. KPConv is one of the best state-of-the-art point cloud segmentation algorithms, which achieves great performance while being robust against varying density neighbourhoods.

PosPool [7] emphasizes also, like the KPConv architecture, on local aggregation computations among the points. It uses a more simple way of computing the local properties of point clouds rather than using complex structures. The authors claim that the PosPool neural network has similar computational capabilities and is more robust to noisy or missing data than most of its competitors, even by using zero learnable parameters for local aggregation operations.

RSConv [6] architecture focus, on local aggregation computations. Equivalent to 2D convolutional operations are used to extract meaningful features from the input point cloud. They are defined by a spherical shaped neighborhood, which is characterized by subsets of input points as centroids. Mainly the point features are derived by a selected relation, for instance the euclidean distance, between the points and the aforementioned centroids. RSConv is considered to be a generalization of the traditional 2D convolutions, as the weights of the convolution matrix are dependent on the position of each point in relation to the center of the convolutional operation. Additionally, it is invariant to permutations, robust to rigid transformations and captures well the relations between the points.

It should be noted that all the aforementioned models evaluate their performance focusing mainly on learning-related metrics, such as *mIoU*.

4.2. Data

We use the ShapeNet dataset for part segmentation, namely ShapeNet-Part[3]. The utilized version of dataset contains 16881 models of 3D point clouds categorized in 16 distinct shapes. In each category, from two to six parts are annotated, summing up to 50 annotated parts in total. The labeled categories are: *aeroplane, bag, cap, car, chair, earphone, guitar, knife, lamp, laptop, motor, mug, pistol, rocket, skateboard and table*. More information on ShapeNet data can be found in its official and published articles [15,16].

4.3. Benchmark Protocol

We have created a benchmark protocol, according to which all the selected models have been trained, validated and tested. The utilized system for the experiments has the following configuration: (i) CPU: Intel Core i9-10900, (ii) RAM: 32GB, (iii) GPU: Quadro RTX 5000 - 16 GB, and (iv) OS: Ubuntu 20.04. In addition, we used the Torch-Points3D [17] framework, using Python 3.8.5, CUDA 10.2 and PyTorch 1.7.0 version.

Regarding the training, validating and testing of the utilized networks, we utilize the original split following the [15]. Specifically, we use a training set size of 12137 point clouds, a validation set size of 1870 and a test set size of 2874. The batch size is set to be 16 and the optimizer of the networks is the Adam. Additionally, to tackle overfitting issues, we use exponential learning rate decay and batch normalization techniques, both of them evaluated at every epoch. Finally, we trained all the networks for 200 epochs. After the training phase on each epoch, the models are validated and tested.

4.4. Results

Figure 1 displays the total run time of each model in relation to the total number of epochs and highlights when the best *CmIoU* and *ImIoU* are recorded in the testing phase of each one of the models. The first observation is that all the models require different run time per epoch to complete the learning process, with the fastest one being the PointNet++ and the slowest one the PPNet. Additionally, the models achieve their best accuracy in different epochs having distinct time duration in their learning processes.

Following our initial observations, we detail the measurements of the learning process of models in Table 1. We observe that RSConv is the best method in test data achieving an *ImIoU* score of 85.47 and a *CmIoU* of 82.73. PointNet++ comes second in terms of *ImIoU* and *CmIoU* metrics, achieving scores of 84.93 and 82.50 respectively. Additionally, PPNet achieves the same *CmIoU* metric of 82.50 as PointNet++.

[3] Available at: https://shapenet.cs.stanford.edu/media/shapenet_part_seg_hdf5_data.zip

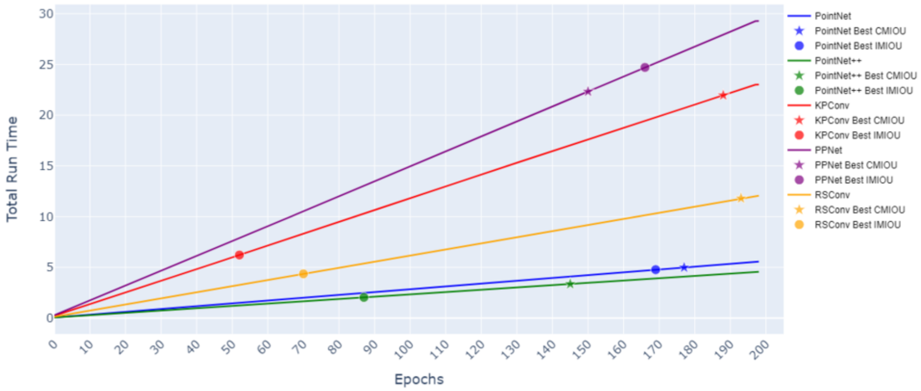


Figure 1. Comparison of Models - Time vs Epochs. The *Total Run Time* is denoted in hours.

Table 1. Deep Learning Models - Performance Evaluation of the learning process in 200 epochs. We display the best achieved *CmIoU* and *ImIoU* evaluation metrics in training, validation and test sets. We further denote the time spent to finish the whole learning process of training, validation and testing as $t_{total}(h)$, the time spent to achieve the best performance in relation to *CmIoU* and *ImIoU* metrics in the test data as $CmIoU^{best}_{time(h)}$, $ImIoU^{best}_{time(h)}$ respectively, and the average GPU memory allocation in percentage in whole learning process as $GPU_{mem}(\%)$. We use dark grey and light grey cell colors to denote the best and the second best score per metric respectively.

Method	**Training Set**		**Validation Set**		**Test Set**		$t_{total}(h)$	$ImIoU^{best}_{time(h)}$	$CmIoU^{best}_{time(h)}$	$GPU_{mem}(\%)$
	ImIoU	*CmIoU*	*ImIoU*	*CmIoU*	*ImIoU*	*CmIoU*				
PointNet	89.80	89.81	86.39	79.87	84.24	79.03	5.55	4.77	4.99	33.7
PointNet++	90.45	90.85	86.83	83.08	84.93	82.50	4.55	2.03	3.36	36.3
KPConv	91.99	91.91	86.61	82.43	84.22	82.39	23.01	6.21	21.96	60.4
PPNet	92.37	92.89	86.46	82.56	83.87	82.50	29.28	24.71	22.33	89.0
RSConv	91.53	92.39	87.12	82.82	85.47	82.73	12.06	4.35	11.81	55.5

An interesting observation is that PointNet++ is the fastest neural network with a total run time (t_{total}) of 4.55 hours, while being the second best in accuracy. It also achieves its peak performance in test data faster than all its competitors, i.e. $ImIoU^{best}_{time(h)} = 2.03$ and $CmIoU^{best}_{time(h)} = 3.36$ hours. RSConv and PointNet come second in $ImIoU^{best}_{time(h)}$ and $CmIoU^{best}_{time(h)}$ respectively.

In terms of average GPU memory usage, PointNet finishes in the first place with 33.7% although the second one, PointNet++, has approximately the same GPU memory usage with a value of 36.3%. Also, PPNet utilizes 89% of total GPU memory while needs about 29.28 hours of time for its computations, appearing to be the most inefficient in terms of memory and time spent. In addition, PPNet reaches the highest performance in learning-related metrics in training set with 92.37 in *ImIoU* and 92.89 in *CmIoU*. However, its poor performance in test set may indicate overfitting issues. Considering our results, the first observation is as follows:

> **Observation 1**. Learning-related *and* system-related *metrics have different winners and the winners of one metric demonstrate lower scores on the others. Thus, a generalized metric that reflects the different behaviour of each model across all scores is needed.*

Table 2. Deep Learning Models - Generalized Performance Evaluation. We use dark grey and light grey cell colors to denote the best and the second best score per metric respectively.

Method	$\beta = 0.3, \alpha = 0.3$			$\beta = 0.5, \alpha = 0.5$			$\beta = 0.8, \alpha = 0.8$		
	F_{ImIoU}	F_{CmIoU}	$F_{general}$	F_{ImIoU}	F_{CmIoU}	$F_{general}$	F_{ImIoU}	F_{CmIoU}	$F_{general}$
PointNet	0.76	0.69	0.72	0.61	0.49	0.55	0.38	0.20	0.29
PointNet++	0.88	0.96	0.92	0.82	0.96	0.89	0.73	0.95	0.84
KPConv	0.34	0.54	0.44	0.30	0.65	0.47	0.25	0.80	0.53
PPNet	0.00	0.28	0.14	0.00	0.47	0.23	0.00	0.75	0.38
RSConv	0.76	0.76	0.76	0.83	0.83	0.83	0.93	0.93	0.93

Our results suggest that certain methodology should be taken into consideration in order to select the best model, according to the use-case's requirements, which is practically accurate but also efficient in time and memory needed for the learning procedure. As shown in Table 1, the most accurate model for part segmentation is the RSConv, according to the *ImIoU* and *CmIoU* metrics on the test dataset. Additionally, we observe that the most efficient model, taking into account the time and GPU memory allocation is PointNet++, while being second in terms of accuracy.

In Table 2 we evaluate our proposed metrics (see section 3.1). We choose to display three different scenarios simulating the final user's needs. In the first scenario, we select values of $\alpha = 0.3$ and $\beta = 0.3$, which indicate that the model selection process will be more biased towards the system-related behaviour of the models. It is clear that PointNet++ architecture achieves the best scores of $F_{ImIoU} = 0.88$, $F_{CmIoU} = 0.96$, $F_{general} = 0.92$. In the second scenario, where $\alpha = 0.5$ and $\beta = 0.5$, we present a balanced approach, giving the same weight in learning- and system-related metrics of each model. In this case, RSConv comes first in F_{ImIoU} metric with a value of 0.83 although the PointNet++ performance is nearly equal with a $F_{ImIoU} = 0.82$. However, PointNet++ outperforms all the other methods in F_{CmIoU} and in $F_{general}$ metrics. Finally, the third scenario, where $\alpha = 0.8$ and $\beta = 0.8$, displays a learning-related model selection process. RSConv seems to be the prime, achieving the best F_{ImIoU} of 0.93 although comes second in F_{CmIoU}. However, according to the generalized metric $F_{general}$, RSConv maintains the highest score of 0.93. Our results raise a second observation:

Observation 2. *A generalized metric facilitates the model selection procedure according to the user's needs, providing a meaningful insight on the trade-off between efficiency and accuracy.*

The three different scenarios presented, show the ability to select the best segmentation model according to an individual's needs. For instance, adjusting the β and α values, to values higher than 0.5 indicates that accuracy is the main concern and values lower than 0.5 indicate that the process concerns efficiency more than accuracy. For a balanced performance evaluation procedure, aiming to select the model that balances accuracy and efficiency, a value of 0.5 should be selected.

To sum up, our initial results confirm, to an extent, the accuracy results, that appear in the literature, for each one of the models analyzed. One important aspect that is derived from our evaluation process is that the learning-related or system-related metrics if used in isolation do not provide clear instruction to a user on which is the best deep learning model. This clearly highlights the need for unified metrics, taking into account both learning- and system-related metrics.

Our findings set us apart from the majority of previous research in this field. While the work in [13,14] presents a time-to-accuracy analysis with a focus on image classification and segmentation tasks, i.e. 2D data, we provide an analysis involving accuracy, run time and memory footprint on point cloud data, i.e. 3D data. Other studies, such as [8], clearly indicate that is of utmost importance to evaluate deep learning segmentation models on the execution time, the memory footprint and the accuracy of them. However, their analysis presents only accuracy results of the utilized models. On the contrary, we analyse all of the aforementioned metrics, providing significant insights.

Actually, examining Figure 1, we could identify some additional remarks, such as **(i)** there is a difference between the epochs where the best $CmIoU$ and $ImIoU$ are obtained, and **(ii)** the PPNet model is the only model of the five that we have investigated, that achieved first the best $CmIoU$ and second the $ImIoU$. Potential next steps are to enrich our proposed metrics by focusing on exploiting this information and taking into account these additional factors and to further analyse the features of each class of the ShapeNet dataset to include the concept of per-point classification accuracy. We expect that the inclusion of all this additional information in our metrics and the rigorous experimentation with more available datasets for the point-cloud part segmentation task may produce new insights into the models.

5. Conclusion

The majority of the research studies in part segmentation analysis using point clouds mostly focus on the improvement of the learning-related metrics of $CmIoU$ and $ImIoU$, i.e. accuracy, and there is a little emphasis on the system-related metrics, i.e. efficiency, and in the trade-off between them. Therefore, the scope of this article has been limited to the analysis of learning-related and system-related performance metrics, which extends considerably the point of view of the current literature in point cloud part segmentation.

This paper provides experimental insights aiming to balance the decision border of the model selection procedure in point cloud part segmentation methodologies. We have analysed five neural network models, namely PointNet, PointNet++, KPConv, PPNet, RSConv, using the mostly utilized ShapeNet dataset in the task of 3D point cloud part segmentation. The first outcome of our analysis is that the most accurate model seems to be the RSConv architecture and the most efficient the PointNet++. We propose the F_{CmIoU}, F_{ImIoU}, and the arithmetic mean of those $F_{general}$. Our metrics aid the benchmarking of different models and provide insights on the trade-off between accuracy and efficiency. We conclude that in a case with a balanced trade-off between accuracy and efficiency, i.e. $\alpha = 0.5$ and $\beta = 0.5$, the selected model should be PointNet++, which is the most efficient while achieves the second highest $CmIoU$ and $ImIoU$ values in test data. Note that our proposed metrics portray the ability to personalize the model selection, by selecting the trade-off between accuracy and efficiency depending on the research objective.

We believe that there are possible further exploration directions, such as the development of a benchmark taking into consideration varying comparison dimensions. As future work, we consider that additional factors should be included in the metrics, such as robustness.

Acknowledgements

Marie Skłodowska-Curie Actions

This project has received funding from the European Unions Horizon 2020 research and innovation programme under the Marie Skodowska-Curie grant agreement No 860843.

References

[1] Bello SA, Yu S, Wang C, Adam JM, Li J. Review: Deep learning on 3D point clouds. MDPI AG; 2020. doi:10.3390/rs12111729.

[2] Wang Q, Tan Y, Mei Z. Computational Methods of Acquisition and Processing of 3D Point Cloud Data for Construction Applications. Archives of Computational Methods in Engineering. 2020 4;27(2):479–499. doi:10.1007/s11831-019-09320-4.

[3] Guo Y, Wang H, Hu Q, Liu H, Liu L, Bennamoun M. Deep Learning for 3D Point Clouds: A Survey. IEEE Transactions on PAMI. 2020:1. doi:10.1109/tpami.2020.3005434.

[4] Nguyen A, Le B. 3D point cloud segmentation: A survey. In: 2013 6th IEEE Conf. on Robotics, Automation and Mechatronics (RAM). IEEE; 2013. p. 225–230. doi:10.1109/RAM.2013.6758588.

[5] Thomas H, Qi CR, Deschaud JE, Marcotegui B, Goulette F, Guibas L. KPConv: Flexible and deformable convolution for point clouds. In: Proceedings of the IEEE Int. Conf. on Computer Vision. vol. 2019-Octob. IEEE Inc.; 2019. p. 6410–6419. doi:10.1109/ICCV.2019.00651.

[6] Liu Y, Fan B, Xiang S, Pan C. Relation-shape convolutional neural network for point cloud analysis. In: Proceedings of the IEEE Computer Society Conf. on CVPR. vol. 2019-June. IEEE Computer Society; 2019. p. 8887–8896. doi:10.1109/CVPR.2019.00910.

[7] Liu Z, Hu H, Cao Y, Zhang Z, Tong X. A Closer Look at Local Aggregation Operators in Point Cloud Analysis. In: Lecture Notes in Computer Science. vol. 12368 LNCS; 2020. p. 326–342. doi:10.1007/978-3-030-58592-1_20.

[8] Garcia-Garcia A, Orts-Escolano S, Oprea S, Villena-Martinez V, Martinez-Gonzalez P, Garcia-Rodriguez J. A survey on deep learning techniques for image and video semantic segmentation. Elsevier Ltd; 2018. doi:10.1016/j.asoc.2018.05.018.

[9] Qi CR, Su H, Mo K, Guibas LJ. PointNet: Deep learning on point sets for 3D classification and segmentation. In: Proceedings - 30th IEEE Conf. on CVPR 2017. vol. 2017-Janua. IEEE Inc.; 2017. p. 77–85. doi:10.1109/CVPR.2017.16.

[10] Qi CR, Yi L, Su H, Guibas LJ. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In: Advances in Neural Information Processing Systems. vol. 2017-Decem. Neural information processing systems foundation; 2017. p. 5100–5109. doi:10.5555/3295222.

[11] Liu W, Sun J, Li W, Hu T, Wang P. Deep learning on point clouds and its application: A survey. MDPI AG; 2019. doi:10.3390/s19194188.

[12] Peyghambarzadeh SMM, Azizmalayeri F, Khotanlou H, Salarpour A. Point-PlaneNet: Plane kernel based convolutional neural network for point clouds analysis. Digital Signal Processing. 2020 3;98:102633. doi:10.1016/j.dsp.2019.102633.

[13] Coleman C, Narayanan D, Kang D, Zhao T, Zhang J, Nardi L, et al. DAWNBench: An End-to-End Deep Learning Benchmark and Competition. 31st Conf on Neural Information Processing Systems (NIPS 2017). 2017. Available from: http://dawn.cs.stanford.edu/benchmark.

[14] Coleman C, Kang D, Narayanan D, Nardi L, Zhao T, Zhang J, et al. Analysis of dawnbench, a time-to-accuracy machine learning performance benchmark. Operating Systems Review (ACM). 2019 7;53(1):14–25. doi:10.1145/3352020.3352024.

[15] Chang AX, Funkhouser T, Guibas L, Hanrahan P, Huang Q, Li Z, et al. ShapeNet: An Information-Rich 3D Model Repository. arXiv. 2015 12. Available from: http://arxiv.org/abs/1512.03012.

[16] Yi L, Kim VG, Ceylan D, Shen IC, Yan M, Su H, et al. A scalable active framework for region annotation in 3D shape collections. ACM Transactions on Graphics. 2016;35(6). doi:10.1145/2980179.2980238.

[17] Chaton T, Chaulet N, Horache S, Landrieu L. Torch-Points3D: A Modular Multi-Task Framework for Reproducible Deep Learning on 3D Point Clouds. In: Proceedings - 2020 Int. Conf. on 3D Vision, 3DV 2020. IEEE Inc.; 2020. p. 190–199. doi:10.1109/3DV50981.2020.00029.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210153

Designing Chest X-ray Datasets for Improving Lung Nodules Detection Through Convolutional Neural Networks

Raúl ACEÑERO EIXARCH [a,1], Raúl DÍAZ-USECHI LAPLAZA [b] and Rafael BERLANGA [a]

[a] *Universitat Jaume I, Castellón, Spain*
[b] *Hospital General de Castellón, Spain*

Abstract. In this paper, we propose a method for building alternative training datasets for lung nodule detection from plain chest X-ray images. Our aim is to improve the classification quality of a state-of-the-art CNN by just selecting appropriate samples from the existing datasets. The hypothesis of this research is that high quality models need to learn by contrasting very clean images with those containing nodules, specially those difficult to identify by non-expert clinicians. Current chest X-ray datasets mostly include images where more than one pathology exist and/or contain devices like catheters. This is because most samples come from old people which are the usual patients subject to X-ray examinations. In this paper, we evaluate several combinations of samples from existing datasets in the literature. Results show a great gain in performance for some of the evaluated combinations, confirming our hypothesis. The achieved performance of these models allows a considerable speed-up in the screening of patients by radiologist.

Keywords. Convolutional Neural Networks, X-ray images, Lung Nodules Detection

Introduction

Radio-diagnosis is a low-cost and universally widespread method based on the analysis of X-ray images. Its main drawback is that it must be carried out by highly qualified people (i.e., radiologists) which are scarce in the public health systems. Thus, most X-ray images are directly delivered to doctors without supervision of radiologists. This motivates the use of automatic screening tools able to detect and send suspicious cases to radiologists. More specifically, these screening system must target to diseases like lung cancer because its early detection is crucial for patients survival.

The initial stage of the neoformative process of lung cancer, before becoming a lung mass, is represented on the chest X-ray as a pulmonary nodule, in most cases in a very subtle way. Even trained eyes can miss these findings, and once the pathology is diagnosed, it is usual that clues become more evident. Lung tumors present different cell lines, and present in different ways in the lung parenchyma. They may have a central

[1] Corresponding Author.

arrangement, being located in perihilar or even endobronchial topographies, as well as a more peripheral distribution. Initially, they will be represented as small pulmonary nodules, always smaller than 3cm, and once that figure is exceeded, they will become pulmonary masses. Densotomographically, they can present densities similar to those of soft tissue tissues. Nodules have poorly defined borders, with a poor definition with respect to the adjacent structures, adopting a morphology of "spiculate borders", which as the disease progresses, can condition the invasion of organs and structures adjacent to the tumor. Given its variability in morphologies and locations where the pulmonary nodule can settle, it is important to recognize its presentation patterns both to identify them and to differentiate them from other pathologies (metastases, abscesses, interstitial diseases and much more).

Currently, there are no automatic methods for early detection of lung tumors in X-ray images, as there are for other neoplasms (breast, prostate, colorectal, etc.) Most methods for nodule detection and classification are defined for computed tomography (CT) scans, which offer greater sensitivity and specificity than X-ray. However, CT scans require high doses of radiation, making the risk/benefit in a large population unfavorable.

1. Related Work

In this section we revise the main approach in dataset generation for automatic radiodiagnosis for lung nodule detection. Most of these approaches rely on state-of-the-art Convolutional Neural Networks (CNN).

In the literature we can find some contributions aimed at generating datasets from existing RIS/PACS systems. Most of them take profit from existing reports attached to the images to automatically generate labels for findings and diseases. For example, ChestX-ray8 [1] provides 108,948 frontal view X-ray images of 32,717 unique patients automatically labeled with eight diseases. Recently, this dataset was extended up to 14 diseases (ChestX-ray14).

ChestX-ray14 was used to train a specialized 121-layers CNN called CheXNet [2]. This model was mainly targeted at detecting pneumonia, but also performed well in the other diseases. Unfortunately, this dataset does not include lung nodules samples.

PadChest[3] provides an even larger dataset with 160,000 images from 67,000 patients labeled with a very large number of concepts automatically extracted from the attached reports. In this case, labels cover 174 findings, 19 differential diagnoses and 104 anatomical locations, which are mapped to the Unified Medical Language System (UMLS). The extraction process is performed by a recurrent neural network trained with 27% of manually annotated images.

As far as we know, there is no work reporting results of a CNN trained with PadChest. Perhaps, an issue for this is to find a method that allow us to get a proper training dataset for the target disease. For example, PadChest contains an acceptable number of samples with lung nodules. However, most samples belong to old people often presenting other pathologies, leading to noisy images. Besides, errors in the image labels can produce further noise in the training datasets.

Compared to ImageNet [6], current chest X-ray datasets do not achieve a critical mass for training CNN models. Most approaches in the literature train and test on the same (small) dataset, without validating the model with other different datasets. More-

CNN model	Dataset	Samples	Classes
ChestX-ray8[1]	NCCIH	112,120	8 diseases (nodules $\approx$ 6,323)
ChestX-ray14[8]	NCCIH	112,120	14 diseases (nodules $\approx$ 6,323)
CheXNet [2]	NCCIH	112,120	14 diseases (nodules $\approx$ 6,323)
Genetic DL[9]	PLCO	185,421	Lung cancer
Faster R-CNN[10]	LIDC-IDRI	1,018	Lung cancer
DENSENET[11]	JSRT	247	Lung nodules
Fastai[12]	Chest X-ray Pneumonia	1,341	Pneumonia/Normal

Table 1. Main CNN approaches and training datasets for nodule detection

over, these approaches show very disparate results, which could indicate the lack of a critical mass in the existing datasets. Thus, current models seem not to generalize well and are not ready yet for production [4]. Authors in [5] discuss the unintended consequences of these drawbacks.

In this short paper we present some preliminary work showing these issues, and how the selection of different samples from different datasets can improve notably the performance of the learned models.

2. Material and Methods

For the screening experiments we mainly use two large datasets of chest X-ray images associated where lung nodules are present. More specifically, we use Chest-X-ray8[7] and PadChest[3]. Table 1 describe these datasets, as well others related to X-ray for extracting samples for the normality class.

Our ultimate goal is the creation of a large and balanced lung cancer dataset that contains X-ray images with nodules difficult to detect by a non-expert people. In this case, images must have associated reports where lung cancer is diagnosed with CT. In addition, X-ray images are requested before the disease is declared, since they will be used to look for patterns in images where the findings are too subtle to be detected by the human eye. For this purpose we added X-ray images from the Hospital of Castelln fulfilling these constraints.

Table 1 shows different CNN approaches along with the datasets with which they were trained on. As earlier mentioned, the choice of these datasets is critical as quality results can vary greatly.

The Tensorflow and Pytorch frameworks have links to their standard neural network zoo models already created along with their precision scores on the Imaginet dataset. One of the CNNs with the highest accuracy rate is Resnet-152, which has been chosen for the experiments.

Resnet-152 was used in two different ways: (1) using a pre-trained model with Imagenet, and (2) training from scratch. In both cases precision results were very similar. However, they notably differ when testing with NCCIH images not seen during the training phase, achieving the models trained from scratch better results. Even so, results were not good enough for screening.

After analyzing the images by a radiologist, we realized that images representing normal cases were not so. Therefore, we decided to look for a series of images completely free from abnormalities. Thus, we selected a Kaggle dataset for pneumonia in children [12]. Training from scratch with a combination of images from NCCIH for nodules and these samples achieved good results for the NCCIH test dataset (100% for nodules and 50% for normal cases).

We tested the new trained model over the PadChest dataset, obtaining good results for the nodule detection but some false positives from images tagged as normal. However, these normal cases belong to old people and contain many noisy elements like electrodes, catheters and so on, which are not reflected in the reports.

3. Conclusions

In this paper we present a preliminary study about the impact of some decisions when training a CNN for classifying X-ray images. Current datasets provided in the literature are either too small or non comprehensive enough to train a deep CNN. Large datasets are provided with tags inferred from the reports, but they do not reflect well the quality and discrimination power of the corresponding images. In our preliminary experiments we show that combining good examples from different datasets can achieve a substantial improvement. In the next steps we aim at building a very large dataset by combining existing ones as well high quality data from hospitals directly assessed by radiologists. The idea is to give the model the necessary elements that allow the model to discern the target pathology at earlier stages. Future work also regards how to automatically discern good from bad examples to train these models.

References

[1] 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 Proc. of CVPR 2017, pp. 3462-3471 (2017)

[2] O. Gozes and H. Greenspan: Deep Feature Learning from a Hospital-Scale Chest X-ray Dataset with Application to TB Detection on a Small-Scale Dataset. In Proc. of EMBC 2019, pp. 4076-4079.

[3] A. Bustos, A. Pertusa, J.M. Salinas, M. de la Iglesia-Vay: PadChest: A large chest x-ray image dataset with multi-label annotated reports. Medical Image Analysis, Vol. 66 (2020)

[4] J. R. Zech, et al: Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study. PLOS Medicine, 15(11):e1002683, 2018.

[5] Federico Cabitza, Raffaele Rasoini, and Gian Franco Gensini. Unintended consequences of machine learning in medicine. Jama, 318(6):517518, 2017.

[6] IMAGENET Dataset, `http://www.image-net.org/`[Last visited: 24/04/2021]

[7] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, R. M. Summers: ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases. IEEE CVPR, pp. 3462-3471 (2017)

[8] Wang H, Jia H, Lu L, Xia Y.: Thorax-Net: An Attention Regularized Deep Neural Network for Classification of Thoracic Diseases on Chest Radiography. IEEE J Biomed Health Inform. Vol. 24(2):475-485 (2020)

[9] A Genome Wide Scan of Lung Cancer and Smoking Dataset. [Last Visited: 20/04/2021] `https://www.ncbi.nlm.nih.gov/projects/gap/cgi-bin/study.cgi?study_id=phs000093.v2.p2/`

[10] A. Bhandary et al.: Deep-learning framework to detect lung abnormality A study with chest X-Ray and lung CT scan images. Pattern Recognition Letters, Vol. 129: 271-278 (2020).

[11] X. Li, et al: Multi-resolution convolutional networks for chest X-ray radiograph based lung nodule detection. Artificial Intelligence in Medicine, Vol. 103 (2020)

[12] Chest X-Ray Images (Pneumonia), [Last visited: 24/04/2021]
`https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia`

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210154

WEU-Net: A Weight Excitation U-Net for Lung Nodule Segmentation

Syeda Furruka BANU [a,1], Md. Mostafa Kamal SARKER [b],
Mohamed ABDEL-NASSER [a], Hatem A. RASHWAN [a], and Domenec PUIG [a]
[a] *Departament d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili, 43007 Tarragona, Spain*
[b] *Precision Medicine Centre of Excellence, The Patrick G Johnston Centre for Cancer Research, Queen's University Belfast, BT9 7BL, United Kingdom*

Abstract. Lung cancer is a dangerous non-communicable disease attacking both women and men and every year it causes thousands of deaths worldwide. Accurate lung nodule segmentation in computed tomography (CT) images can help detect lung cancer early. Since there are different locations and indistinguishable shapes of lung nodules in CT images, the accuracy of the existing automated lung nodule segmentation methods still needs further enhancements. In an attempt towards overcoming the above-mentioned challenges, this paper presents WEU-Net; an end-to-end encoder-decoder deep learning approach to accurately segment lung nodules in CT images. Specifically, we use a U-Net network as a baseline and propose a weight excitation (WE) mechanism to encourage the deep learning network to learn lung nodule-relevant contextual features during the training stage. WEU-Net was trained and validated on a publicly available CT images dataset called LIDC-IDRI. The experimental results demonstrated that WEU-Net achieved a Dice score of 82.83% and a Jaccard similarity coefficient of 70.55%.

Keywords. Lung Cancer, Computed Tomography (CT), Lung Nodule Segmentation, Computer-Aided Diagnosis (CAD), Deep Learning.

1. Introduction

Lung cancer causes around 1.3 million deaths each year over the world according to the World Health Organization (WHO). The death rate in European Union (EU) is about 44%, in turn in Japan is 35%. Since lung cancer detection is a challenging task in early stages; most cases are frequently diagnosed in the late-stage, and thus patients cannot survive because of improper prediction of disease and treatment at the late-stage of the disease at the hospital.The accurate detection and analysis of the lung nodules in the lung tissue in an early phase highly increase the patient chance of survival and facilitates effective treatment [1]. Computed tomography (CT) scans are a popularly utilized and profoundly accurate format for analyzing the lung nodules.The multi-detector row CT scanners are utilized to collect these lung scans.The precise segmentation of the lung nodules has a great impact on the diagnosis and prognosis of lung cancer [2]. Therefore,

[1] Corresponding Author: E-mail: syedafurruka.banu@estudiants.urv.cat

a radiologist must go through all individual patient CT scans that contain so many slices (about 150-500), which is a very difficult and tedious task [3]. However, it is difficult to distinguish the internal lung structure and the nodules because of the complex tissue environment, particularly when the nodule is located on the lung surface or attached to the border of a vessel in the lung tissue. To an accurate segmentation of the lung nodules by using normal image processing technique (e.g., threshold and morphological-based methods) is very challenging due to the large variety in size and types of lung nodules [4]. Another, the segmentation of nodules with small diameter and intensity variations with the surrounding noise is also very difficult for segmenting the nodule accurately. Thus, it is necessary to develop a robust segmentation system that can adapt to all the limitations and achieves efficient performance. Recently, convolutional neural networks (CNN) have become dominant in the area of computer vision (e.g., semantic segmentation, image classification, object detection, etc.). In the biomedical image analysis domain, an encoder-decoder like CNN architecture called U-Net [5], shown exceptional results on the task of segmentation. Several modified version of the U-Net yields state-of-the-art results in this domain. Nonetheless, the development of CNN architectures for the segmentation of lung nodules is yet immature. Hence, it is necessary to develop advanced architectures that can deal with the weaknesses of the previous architectures. In this paper, a weight excitation (WE) mechanism through the weight reparameterization-based backpropagation corrections is used from [6] and implemented with the U-Net encoder and decoder architecture to cope with the heterogeneity of lung nodule feature extraction efficiently, which is suitable for the segmentation of various forms of lung nodules.The main contributions of this research can be summarized as follows:

- We propose WEU-Net, which is a robust and efficient lung nodule segmentation model.
- A weight excitation (WE) mechanism through the weight reparameterization-based backpropagation corrections is adapted with the U-Net.
- The WEU-Net model achieved an overall precise segmentation performance on different challenging cases of nodule segmentation.

The preparation of this article is as follows. Section II discusses recent lung nodules segmentation methods based on traditional methods and deep learning techniques. The details of the proposed models architecture and the experimental setup and results are described in Section III and IV, respectively. Finally, Section V concludes and proposes some future works of this research.

2. Related Works

Several traditional (e.g., morphological, region-growing, and energy-based techniques, etc.), machine learning and deep learning-based lung nodule segmentation approaches that have been introduced in the last few years are described in this section. In traditional approaches, a morphological based-operation to remove the nodules associated with the vessels, and isolating the lung nodules by choosing the connected region is introduced in [7]. Another improvement of the separation of nodules from the lung surface is presented in [8] by using the combination of morphological based-method with the shape condition. However, the segmentation of lung nodules by morphological operations is extremely challenging [9]. [4] presented that other well-known region-growing methods are not able to segment the nodules perfectly among the different types. In [10] noticed these challenges and proposed a region-growing technique based on the intensity information, distance, fuzzy connectivity, and peripheral variation of the nodules. The challenge of these tasks is the convergence condition of the methods. Furthermore, it is also hard to segment the irregular and fuzzy-shaped nodules using the region-growing techniques because of the nodule shape condition. Moreover, the energy based-optimization methods are also introduced to segment the lung nodules by [11], [12], [13], [14] using a level set function to minimize the energy function for achieving the segmented contour which reaches the boundary of the lung nodule. In [15]; [16]; [17] presented a maximum flow concept similar to the region-growing methods for segmenting the low contrast nodules.

Over the last decade, researchers are introduced machine-learning approaches for the classification, segmentation, and detection of lung nodules. In [18] presented extraction of rich feature maps with the invariance of translation and rotation. Other textures and shape-dependent features are utilized for the classification of voxels of lung nodules with conditional random field (CRF) model is presented in [19]. The Hessian matrix-based vascular feature extraction procedure is introduced in [20]. and used lung blood vessel segmentation and classification. Another Hessian-based 3D large-scale nodule segmentation method is utilized by [21]. Currently, deep learning approaches outperform the traditional and machine learning-based methods in most of the domains. Several CNN-based models were proposed for the task of the classification and segmentation of voxels in a supervised manner. For example, the multi-view convolutional neural network (MVCNN) is proposed by [22]. for nodule segmentation, which consists of three branches of CNN modules that are related to the three positions of the sagittal, axial, and coronal plane. Familiarly, fully convolutional networks (FCN), 2D U-Net, and 3D U-Net architectures are introduced by [23], [5], and [24], respectively are commonly used deep learning models for the segmentation of the biomedical imaging. Similarly, the central focused convolutional neural network(CF-CNN) is based on the shape-aware method proposed by [25], for the lung nodules segmentation task. Recently, a dual-branch residual network (DBResNet) is proposed by [26] using the combination of intensity features into the CNN model to achieve better segmentation results on lung nodules.

3. Proposed Model

The proposed WEU-Net is a encoder-decoder model shown in Figure 1 that modified version of the popular U-Net [5] model for the biomedical image segmentation task. The encoder part of the WEU-Net is composed by weight excitation based CNN (WE-CNN) [6], ReLU activation and MaxPooling layers. The input size of the model is $256x256x1$, because of the single channel CT lung nodule image. Initially, the WE-CNN and ReLU layers are apply to capture the contextual features from the input image. The encoder is a collection of WE-CNN and max pooling layers. The size of the feature maps are gradually decreases while the depth successively increases in the encoder from $256x256x64$ to $16x16x1024$. The decoder is the stack of transposed WE-CNN layers to up-sample the feature maps to the same size of the model input. The size of the decoder features are gradually increases and the depth gradually decreases from $32x32x512$ to $256x256x2$.

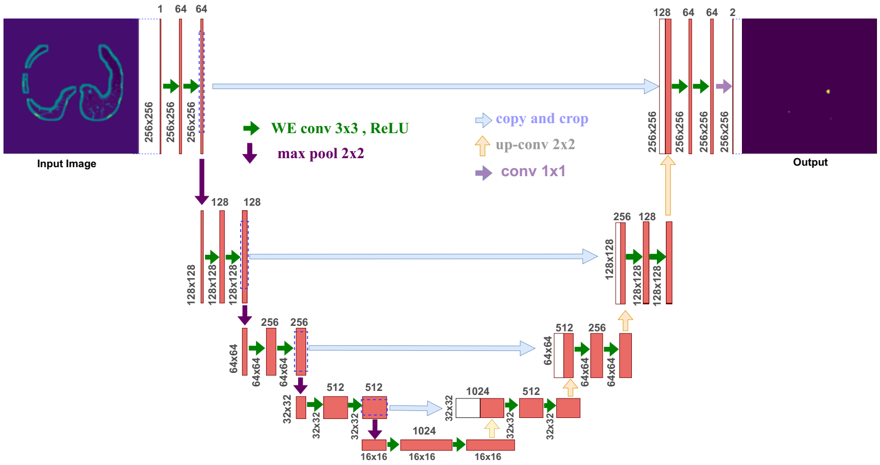


Figure 1. The architecture of the proposed WEU-Net.

After every decoder block, the feature maps are up-sampled and get the same size as the corresponding encoder block output to maintain harmony and concatenated it. This mechanism helps to keep the features that are learned from the encoder phase and use them for the reconstruction method. A 1×1 WE-CNN is used at the final layer of the network to map the final 64 feature vector to the targeted number of the segmentation classes which is two (background and lung nodule). A total of 23 WE-CNN layers are used in the U-Net model.

4. Experimental Setup and Results

4.0.1. Dataset

We used a publicly available dataset from the Lung Image Database Consortium and Image Database Resource Initiative (LIDC-IDRI) [27] to train and evaluate the proposed WEU-Net. The dataset consists of 888 paitents CT scans with annotations that were annotated by four experienced radiologists by using two-phase annotation process. The dataset contains annotations of 1186 nodules with the annotation files indicating different features of the nodules. The dataset remains with a total of 1166 CT images with corresponding ground truth masks after the pre-processing method is used for cleaning the dataset. We then split the dataset into two subsets training and test with 922 and 244 CT images, respectively.

4.0.2. Evaluation Metrics

We evaluate the proposed WEU-Net by using five metrics, accuracy (ACC), intersection over union (IoU), Dice similarity coefficient (DIC), precision (PRE), and recall (REC). Let the annotated ground-truth is y, the true positive (TP) and the false positive (FP) rates are correctly and incorrectly classified pixels as y, whereas the true negative (TN) and the false negative (FN) rates are correctly and incorrectly classified pixels as not y. The mathematical definitions of the five metrics: ACC, IoU DIC, PRE, and REC are presented following.

$$ACC = \frac{TP+TN}{TP+TN+FP+FN} \tag{1}$$

$$IoU = \frac{TP}{TP+FP+FN} \tag{2}$$

$$DIC = \frac{2.TP}{2.TP+FP+FN} \tag{3}$$

$$PRE = \frac{TP}{TP+FP} \tag{4}$$

$$REC = \frac{TP}{TP+FN} \tag{5}$$

4.0.3. Data Augmentation and Implementation

We used online data augmentation methods during the training phase of the proposed model to increase the amount of training data by flipping the images horizontally and vertically, random cropping, and rotating. We implemented the proposed model on the PyTorch framework [28] and used NVIDIA 1080Ti GPU with 8GB memory. The Adam optimizer [29] is used with the learning rate and the batch size are set to 0.0002 and 8, respectively. The binary cross-entropy with dice loss is used as a loss function to train the proposed model.

4.0.4. Results

We carried a comprehensive investigation to evaluate how our proposed WEU-Net model improves the segmentation performance compared to the four state-of-the-art semantic segmentation models, PSPNet [30], Linknet [31], FPN [32], and vanila U-Net [5]. A performance comparison between the proposed model and four segmentation methods on the LIDC-IDRI dataset presents in Table 1.

Table 1. A comparison between the proposed model and four segmentation methods on the LIDC-IDRI dataset

Model Name	Metrices				
	ACC	IoU	DIC	PRE	REC
PSPNet [30]	99.87	62.12	77.34	73.00	83.78
Linknet [31]	99.89	63.30	77.43	75.00	83.58
FPN [32]	99.91	64.77	78.56	78.62	80.81
U-Net [5]	99.95	67.70	80.49	81.21	82.28
WEU-Net	**99.99**	**70.5**5	**82.83**	**81.66**	**86.53**

The proposed model achieved the highest performance of 99.99%, 70.55%, 82.83%, 81.66%, and 86.53% in terms of ACC, IoU, DIC, PRE, and REC, respectively. Comparing to vanila U-Net, it improves 0.04%, 2.85%, 2.34%, 0.45%, and 4.25% of ACC, IoU, DIC, PRE, and REC, respectively. Therefore, it clearly shows that the effect of the weight excitation-based CNN is significantly improving the performance of U-Net. On the other hand, it also improves the IoU of 8.43%, 7.25%, 5.78%, and 2.85% and the DIC of 5.49%, 5.40%, 4.27%, and 2.34% compared to the PSPNet, Linknet, FPN, and vanila U-Net, respectively.

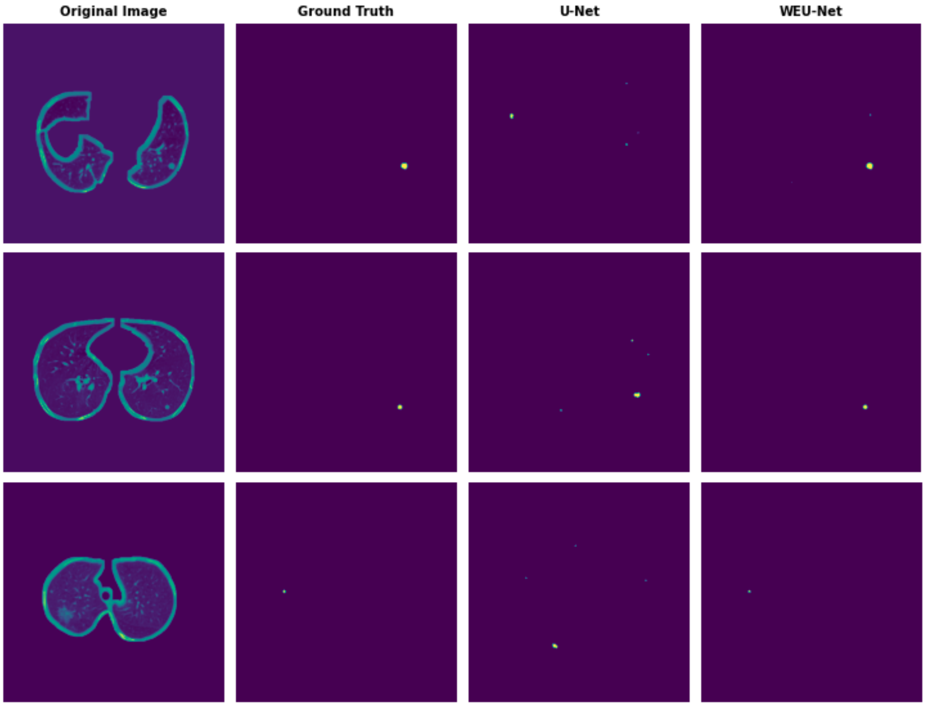


Figure 2. Examples of segmentation results (from left to right column represents; original image, ground truth, segmented by U-Net and proposed WEU-Net).

Figure 2 illustrates the qualitative segmentation results of WEU-Net from some examples of the LIDC-IDRI dataset. As can be seen in the predictions of U-Net model contains wrong segmentation and consists of false positive, whereas the proposed model segment the nodule region accurately. The proposed model can also segment the tiny nodule region preciously when the vanila U-Net failed to segment it. Therefore, the visualization demonstrates that the proposed model improved the U-Net model performance significantly for the lung nodule segmentation task.

5. Conclusion

This paper proposes a weight excitation U-Net for lung nodule segmentation. The weight excitation mechanism extracts the features in a weighted manner which leads to improving the performance of the vanila U-Net. The experimental evaluation shows that the proposed model demonstrated promising improvement compared with the four state-of-the-art semantic segmentation models including vanila U-Net. The proposed model achieved 99.99%, 70.55%, 82.83%, 81.66%, and 86.53% of ACC, IoU, DIC, PRE, and REC, respectively on the LIDC-IDRI dataset. The future work focuses on developing a

3D WEU-Net model based on the 3D weight excitation-based CNN (3D WE-CNN) for fully automated segmentation and classification of lung cancer.

Acknowledgement

The Spanish Government partly supported this research through Project PID2019-105789RB-I00.

References

[1] A. El-Baz and J.S. Suri, *Lung imaging and computer aided diagnosis*, CRC Press, 2011.

[2] H. MacMahon, J.H. Austin, G. Gamsu, C.J. Herold, J.R. Jett, D.P. Naidich, E.F. Patz Jr and S.J. Swensen, Guidelines for management of small pulmonary nodules detected on CT scans: a statement from the Fleischner Society, *Radiology* **237**(2) (2005), 395–400.

[3] T. Way, H.-P. Chan, L. Hadjiiski, B. Sahiner, A. Chughtai, T.K. Song, C. Poopat, J. Stojanovska, L. Frank, A. Attili et al., Computer-Aided Diagnosis of Lung Nodules on CT Scans:: ROC Study of Its Effect on Radiologists' Performance, *Academic radiology* **17**(3) (2010), 323–332.

[4] T. Kubota, A.K. Jerebko, M. Dewan, M. Salganicoff and A. Krishnan, Segmentation of pulmonary nodules of various densities with morphological approaches and convexity models, *Medical Image Analysis* **15**(1) (2011), 133–154.

[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*, Springer, 2015, pp. 234–241.

[6] N. Quader, M.M.I. Bhuiyan, J. Lu, P. Dai and W. Li, Weight Excitation: Built-in Attention Mechanisms in Convolutional Neural Networks, in: *European Conference on Computer Vision*, Springer, 2020, pp. 87–103.

[7] W.J. Kostis, A.P. Reeves, D.F. Yankelevitz and C.I. Henschke, Three-dimensional segmentation and growth-rate estimation of small pulmonary nodules in helical CT images, *IEEE transactions on medical imaging* **22**(10) (2003), 1259–1274.

[8] D. Sargent and S.Y. Park, Semi-automatic 3D lung nodule segmentation in CT using dynamic programming, in: *Medical Imaging 2017: Image Processing*, Vol. 10133, International Society for Optics and Photonics, 2017, p. 101332.

[9] S. Diciotti, S. Lombardo, M. Falchini, G. Picozzi and M. Mascalchi, Automated segmentation refinement of small lung nodules in CT scans by local shape analysis, *IEEE Transactions on Biomedical Engineering* **58**(12) (2011), 3418–3428.

[10] J. Dehmeshki, H. Amin, M. Valdivieso and X. Ye, Segmentation of pulmonary nodules in thoracic CT scans: a region growing approach, *IEEE transactions on medical imaging* **27**(4) (2008), 467–480.

[11] T.F. Chan and L.A. Vese, Active contours without edges, *IEEE Transactions on image processing* **10**(2) (2001), 266–277.

[12] P.P. Rebouças Filho, A.C. da Silva Barros, J.S. Almeida, J. Rodrigues and V.H.C. de Albuquerque, A new effective and powerful medical image segmentation algorithm based on optimum path snakes, *Applied Soft Computing* **76** (2019), 649–670.

[13] E.E. Nithila and S. Kumar, Segmentation of lung nodule in CT data using active contour model and Fuzzy C-mean clustering, *Alexandria Engineering Journal* **55**(3) (2016), 2583–2588.

[14] A.A. Farag, H.E. Abd El Munim, J.H. Graham and A.A. Farag, A novel approach for lung nodules segmentation in chest CT using level sets, *IEEE Transactions on Image Processing* **22**(12) (2013), 5202–5213.

[15] X. Ye, G. Beddoe and G. Slabaugh, Automatic graph cut segmentation of lesions in CT using mean shift superpixels, *International journal of biomedical imaging* **2010** (2010).

[16] S. Mukherjee, X. Huang and R.R. Bhagalia, Lung nodule segmentation using deep learned prior based graph cut, in: *2017 IEEE 14th international symposium on biomedical imaging (ISBI 2017)*, IEEE, 2017, pp. 1205–1208.

[17] Y. Boykov and V. Kolmogorov, An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision, *IEEE transactions on pattern analysis and machine intelligence* **26**(9) (2004), 1124–1137.

[18] L. Lu, P. Devarakota, S. Vikal, D. Wu, Y. Zheng and M. Wolf, Computer aided diagnosis using multilevel image features on large-scale evaluation, in: *International MICCAI Workshop on Medical Computer Vision*, Springer, 2013, pp. 161–174.

[19] D. Wu, L. Lu, J. Bi, Y. Shinagawa, K. Boyer, A. Krishnan and M. Salganicoff, Stratified learning of local anatomical context for lung nodules in CT images, in: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, 2010, pp. 2791–2798.

[20] Y. Hu and P.G. Menon, A neural network approach to lung nodule segmentation, in: *Medical Imaging 2016: Image Processing*, Vol. 9784, International Society for Optics and Photonics, 2016, p. 97842.

[21] L. Gonçalves, J. Novo and A. Campilho, Hessian based approaches for 3D lung nodule segmentation, *Expert Systems with Applications* **61** (2016), 1–15.

[22] J. Wang and H. Guo, Automatic approach for lung segmentation with juxta-pleural nodules from thoracic CT based on contour tracing and correction, *Computational and mathematical methods in medicine* **2016** (2016).

[23] J. Long, E. Shelhamer and T. Darrell, Fully convolutional networks for semantic segmentation, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[24] Ö. Çiçek, A. Abdulkadir, S.S. Lienkamp, T. Brox and O. Ronneberger, 3D U-Net: learning dense volumetric segmentation from sparse annotation, in: *International conference on medical image computing and computer-assisted intervention*, Springer, 2016, pp. 424–432.

[25] S. Wang, M. Zhou, O. Gevaert, Z. Tang, D. Dong, Z. Liu and T. Jie, A multi-view deep convolutional neural networks for lung nodule segmentation, in: *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, 2017, pp. 1752–1755.

[26] H. Cao, H. Liu, E. Song, C.-C. Hung, G. Ma, X. Xu, R. Jin and J. Lu, Dual-branch residual network for lung nodule segmentation, *Applied Soft Computing* **86** (2020), 105934.

[27] S.G. Armato III, G. McLennan, L. Bidaut, M.F. McNitt-Gray, C.R. Meyer, A.P. Reeves, B. Zhao, D.R. Aberle, C.I. Henschke, E.A. Hoffman et al., The lung image database consortium (LIDC) and image database resource initiative (IDRI): a completed reference database of lung nodules on CT scans, *Medical physics* **38**(2) (2011), 915–931.

[28] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai and S. Chintala, PyTorch: An Imperative Style, High-Performance Deep Learning Library, in: *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox and R. Garnett, eds, Curran Associates, Inc., 2019, pp. 8024–8035. `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf`.

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

[30] H. Zhao, J. Shi, X. Qi, X. Wang and J. Jia, Pyramid scene parsing network, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881–2890.

[31] A. Chaurasia and E. Culurciello, Linknet: Exploiting encoder representations for efficient semantic segmentation, in: *2017 IEEE Visual Communications and Image Processing (VCIP)*, IEEE, 2017, pp. 1–4.

[32] A. Kirillov, K. He, R. Girshick and P. Dollár, A unified architecture for instance and semantic segmentation, 2017.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210155

Affective State-Based Framework for e-Learning Systems

Juan Antonio RODRÍGUEZ [a], Joaquim COMAS [a] and Xavier BINEFA [a]
[a] *Department of Information and Communication Technologies, Pompeu Fabra University, Barcelona, Spain*

Abstract. Virtual learning and education have become crucial during the COVID-19 pandemic, which has forced a rethink by teachers and educators into designing online content and the indirect interaction with students. In an face-to-face class, some visual cues help the teacher recognize the engagement level of students, while the main weakness of the online approach is the lack of feedback that the teacher has about the learning process of the students. In this paper, we introduce a novel framework able to track the learning states, or LS, of the students while they are watching a piece of knowledge-based content. Specifically, we extract four learning states: Interested, Bored, Confused or Distracted. Finally, to demonstrate the system's capability, we collected a reduced database to analyze the affective state of the subjects. From these preliminary results, we observe abrupt changes in the LS of the audience when there are abrupt changes in the narrative of the video, indicating that well-structured and bounded information is strongly related with the learning behaviour of the students.

Keywords. Learning states, e-Learning, Deep Learning, Machine Learning, Facial Expression Analysis

1. Introduction

Online education has become very popular due to lockdown measures for the COVID-19 pandemic, and the increase of online resources in the form of MOOCS. MOOCS are very specialised courses based on any subject, that allow students to work asynchronously from any location using smart devices. However, this paradigm provokes a huge disconnection between the learner and the teacher, as the former one loses the ability of adapting the session based on the emotional reception of the students. Future E-Learning systems must integrate mechanisms to predict learning states of the students in order to drive the learning session and enhance the educational experience.

Affective Computing is the field that studies how machines will be able to assist humans or make decisions based on emotional information. The term was first introduced by Picard [1], where she also stated how E-Learning systems may integrate affective capabilities to keep the students engaged. This problem can be tackled using many sources of data in a multi-modal manner, using physiological signals like heart rate (HR), skin conductance response (SCR), blood volume pulse (BVP) and electroencephalogram (EEG) as well as facial, body pose images or audio signals. Any biological signal can correlate with the affective state of a person, however the non-invasive ones must be our choice in order to build easy-to-use systems and put users' attention in the learning part. Following the last argument, most of the systems and models for E-Learning and Emotion Prediction use Computer Vision techniques for Facial Expression Analysis and the extraction of other visual cues. Nevertheless multi-modal fused models show superiority in accuracy [2] over just Facial Expression Analysis, at the cost of having more complex set-ups.

There has been a lot of efforts to find the best prediction/target models in order to assess emotional states. The Valance-Arousal two dimensional space introduced by Russell [3], is the choice of many works, as it encodes emotional states in two dimensions and gives a mapping to discrete classes. Other approaches use a classification model with classes of emotions focused in the task of choice. For E-Learning, classes may come from the set of Engagement, Interest, Confusion, Boredom, Distraction or Frustration. Some works may also use Emotion Levels bounded from 0 to 1 for a given class, and solve the problem by means of regression. With the explosion of Deep Learning and Neural Networks, many prediction tasks have been solved with huge accuracy and robustness. This is due to the effort on building larger models in terms of parameters, better training mechanisms and also the increment of training data available. New technologies allow for the integration of such models into smart devices in a very optimized manner, contributing for accuracy and speed. This work in progress focuses on studying many deep learning models for E-Learning Emotion Prediction and effectively use this functionality to enhance the E-Learning experience. Experiments in this paper will use a baseline model based on Convolutional Neural Networks (CNN), to focus on the system's emotion mechanics and analysis.

In this work, we focus on a specific type of resource in a learning session, which is the video. Almost every online course is supported by video based content to explain important information from a topic. These videos must be precisely designed, structured and narrated in order to achieve the goal of maintaining the student interested and transmitting the desired knowledge. We propose a system that predicts learning states from facial expressions, and links this result with a time instant of the video. With this synchronisation we can assess the evolution of emotions through the narrative of the video and also analyze if the narrative structure is properly designed.

The rest of the paper is structured as follows: Section 2 overviews literature and related work on the problem of affective state-based E-Learning. In Section 3 we describe the methodology of building the video-based E-Learning system, with details on the E-learning workflow, the prediction model and the datasets for training. In Section 4 we report preliminary experiments and results. In section 5 we reflect on ethics, fairness and privacy of works of this kind, when sensitive data is gathered from users and biased prediction models will most likely appear. Finally, Section 6 explains conclusions and future work.

2. Related Work

The task of Affect Recognition in E-Learning scenarios has shown challenges in both the methodology used for predicting the Learning States from raw data, and the design of plausible system models capable of using this predictions in benefit of the learning experience. The following is an overview of studies that focus on those two problems.

One of the earliest uses of physiological data for automatic emotion estimation on e-learning environments has been the work of Shen et al. [4], using signals such as HR, SCR, BVP and EEG. The authors proposed Support-Vector Machine (SVM) as prediction model in the Valence-Arousal space. In this early work, it is shown how emotion aware systems increment the interaction and engagement of the subjects.

Porta et al. [5] presented a preliminary study of a system able to adapt to the users learning states through eye tracking data. They find that variation in pupil size is highly correlated with users actions and mental effort.

Also, Ashwin et al. [6] conducted experiments in a face-to-face system focused on efficiently detect facial images from multiple users, and predict seven emotions using SVM with some well-known datasets.

Alyuz et al. [7] relied on an appearance data gathered from the camera and context information from the platform, and proposed a method with a calibration phase in order to personalize the model to the subject. Their architecture predicted Satisfaction, Boredom and Confusion using a self-collected dataset and Random Forest method.

The work presented by Chen et al. [2] introduce a hybrid model combining several modalities based on facial and physiological cues, to predict valence, arousal and attention. The authors propose a model for assisting the user based on those three indicators, using SVM and multi-modal fusion.

Luo et al. [8] discuss a new model for affective learning based in three dimensions: Attention, Emotion and Thinking. In this work, experiments where conducted using only facial images in a face-to-face e-Learning session, capturing facial and emotional data from many subjects concurrently. They use classical methods and fusion to obtain a final indicator of interest.

In the recent literature, other data-driven methods have introduced more sophisticated models and provided dedicated datasets to assess the emotional state in e-learning systems. First example is the work of Kaur et al. [9] which introduces a new dataset with collected videos and eye gaze data from students in a learning set-up, proposed for the task of Engagement Level Prediction. The authors provide many baseline models and results using modern deep learning networks.

Following the line of works in the deep learning paradigm, the use of CNN for image feature extraction has become the standard choice. Emotion Recognition studies conducted in [10], [11], [12], [13], [14] achieve State of the art accuracy on very popular datasets using CNN using both classification and regression targets.

Finally, Mukhopadhyay et al. [15] proposed a CNN-based method to assess the state of mind of the learners during an online learning session extracting Learning States based in the combinations of the eight basic emotions, namely confusion, dissatisfaction, satisfaction, and frustration.

A limitation for the CNN baseline architecture arises when images are from a video source, and vary in the time dimension. In order to reach high accuracy for spatio-temporal data patterns one must set-up a training and inference mechanic to treat data in a sequence-like fashion. In that sense, there is a lack of studies that focus on this type of data structures for Affective E-Learning systems.

The main contribution of this paper is the design and preliminary test of a novel framework for e-learning, that allows educators to obtain feedback about the LS of the students during the visualization of a video. The main aspects are the following:

1. Propose a web application-based E-Learning system focused on the visualization of video resources. The presented framework links video timestamps with LS predictions to asses the quality of the video design and the affective evolution of the student.
2. Show performance and accuracy results of a baseline data-driven model optimized for performance in edge devices. The model uses CNN feature extraction and classification.

3. Methodology

3.1. Experimental set-up and recording

The designed system can be accessed through a web application with a computer or mobile device, and the student must agree and give consent of data treatment, see part 4 Ethics, fairness and privacy. At the start of the session, the user has many video options to visualize, which are knowledge-based to introduce the student into a learning challenge. Afterwards, the system warms up, loading both face detection and learning state prediction models, as well as camera recording.

When the user feels ready, starts the visualization, and the system begins to detect faces at a chosen frame rate, which are then cropped and resized to be propagated through the neural network. The raw predictions, facial images and time records are stored in the database for further analysis.

One of the key functionalities of the system is *Ground Truth Feedback* or GTF, a mechanism that lets the teacher define questions in relevant parts of the narrative structure of the video that will be prompted to the student. This information is stored to ease the testing and validation of both the training and accuracy of the deep learning models, assess the learning process of the students, and the quality of the video.

When the visualization is finished, the recorded video of the subject is stored in the database for training purposes. The full video recording will allow further researchers to explore new ways of processing the data, having all frames and more data modalities such as head or body pose.

Algorithm 1 describes the process, and it consists of a loop over the whole video sequence, where the operations of face extraction, LS prediction and storage are performed subsequently. At every iteration, the system checks if it needs to ask a feedback question based on a list of time samples generated by the teacher. As the questions may be different, the current video time is passed as a parameter. Parallel to this process, the system is recording the camera video stream, which will be encoded in *H.264* and stored in the database as a *.webm* file. These videos will contribute to the *Learning Students Dataset*, alongside the automatic predictions and the GTF.

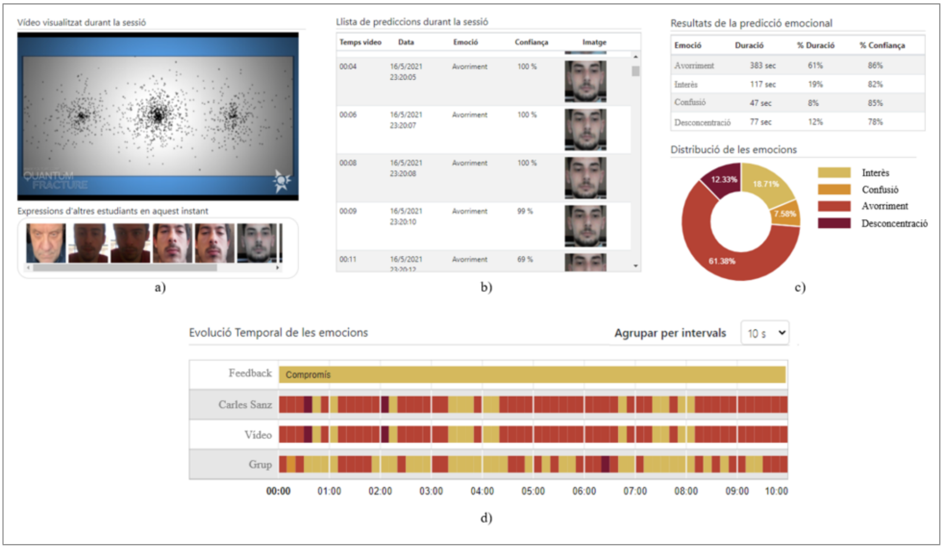
After the learning session, the subject moves to the *Course Analyzer*, Figure 1, an interactive dashboard that allows the learner to analyze the session in terms of the LS evoked by the video. The *Course Analyzer* generates charts that show the distribution of the learning state predictions over time, which gives an indicator of the overall reactions to the content.

We also generate a detailed table with the stored LS and facial expressions, which by clicking over, the system plays the video in that instant of time as well as shows facial expressions of other participants at that time. This gives the teacher or content creator a tool to analyze how the narrative is being received.

```
Data: YouTube video, webcam stream
Result: LS, facial images, GTF
while video playing do
    if feedbackTime(videoTime) then
        GTF = askFedback(videoTime)
        store(GTF)
    end
    faceImage = camera.detectFace()
    if faceImage then
        LS = nn.prediction(faceImage)
        storeData(LS, faceImage, videoTime, timestamp)
    end
end
storeFullVideo()
```

Algorithm 1: Workflow of the learning session.

Finally, we define the *Learning Timeline*, a chart that aggregates the LS over windows of time intervals. Over these intervals, a Majority Voting Rule (MVR) is computed, which returns just one LS with information also from the time domain. The *Learning Timeline* also shows the answers given by the student in the GTF. This is computed at three levels, student level, for one learning session, video level, for many visualizations of a user, and group level, for many students in the same video.


Vídeo visualitzat durant la sessió
QUANTUM FRACTURE
Expressions d'altres estudiants en aquest instant
a)
b)
c)
Evolució Temporal de les emocions
Feedback
Compromís
Carles Sanz
Video
Grup
00:00
01:00
02:00
03:00
04:00
05:00
06:00
07:00
08:00
09:00
10:00
d)

Llista de prediccions durant la sessió

Temps video	Data	Emoció	Confiança	Imatge
00:04	16/5/2021 23:20:05	Avorriment	100 %	
00:06	16/5/2021 23:20:07	Avorriment	100 %	
00:08	16/5/2021 23:20:08	Avorriment	100 %	
00:09	16/5/2021 23:20:10	Avorriment	99 %	
00:11	16/5/2021 23:20:12	Avorriment	69 %	

Resultats de la predicció emocional

Emoció	Duració	% Duració	% Confiança
Avorriment	383 sec	61%	86%
Interès	117 sec	19%	82%
Confusió	47 sec	8%	85%
Desconcentració	77 sec	12%	78%

Distribució de les emocions

18.71%	Interès
7.58%	Confusió
61.38%	Avorriment
12.33%	Desconcentració

Agrupar per intervals 10 s

Figure 1. Course Analyzer. a) Video visualized by the student and facial expressions of other students in that instant. b) Learning state predictions in detail. c) Distribution of the learning states. d) Learning Timeline at three levels and feedback answers of the student.

3.2. Proposed method

We use data driven-based inference in order to make predictions of the LS from facial images. We use the *MobileNetV2* architecture [16] as a backbone network that is specially designed for working on edge devices efficiently. This model is based on Convolutional Neural Networks and Inverted Residual structures, which allow the network to learn powerful low-dimensional representations at a small computational cost.

This framework acts as a baseline for predicting the LS, which can effectively model the space domain of the data, i.e. it processes the images frame by frame, not taking into account the sequential nature of the data. Using the Majority Voting Rule over a time interval, the framework can give a final prediction using the time domain at low cost. However, this baseline architecture is not able to use the time domain in the training stage. For that purpose, we need more sophisticated architectures that do the prediction task using a sequence of frames as input. As a work-in-progress, we are analysing the impact of Temporal Convolutional Networks (TCN) [17]. These new models will come at a more computational cost but may exhibit more accurate and robust performance.

3.3. Data and Training

The training data consists of facial images paired with the annotated learning state class that the subject is evoking. The source of this data is a combination of public datasets, namely Affectnet, CK+ and DAiSEE. Affectnet dataset [18] consists of more than 1M images of facial expressions annotated with eight basic emotions: Disgust, Happy, Surprise, Fear, Angry, Contempt, Sadness and Neutral. CK+ dataset [19] is formed by 593 video sequences showing facial expressions of people, also annotated with the basis of eight emotions. CK+ gives significant amounts of data of not-in-the-wild examples, which means that the conditions of the images and facial expressions are very staged. These datasets do not contain labels for our task, but we performing a training stage for classification with the eight classes, for further transfer of learning.

Furthermore, we apply transfer learning to the model with the four LS. This process is done by freezing the weights of the first convolutional layers, and training the model with the new labels until convergence. The second training was done using the DAiSEE Dataset [20], which contains over 9000 video sequences of students in an e-Learning session. They are labelled using four labels that represent the levels of the e-Learning emotions: Boredom, Engagement, Confusion and Frustration. We performed a supervised re-annotation of the videos in order to get our four LS. This process consisted in labelling facial frames with one of the four emotions with a level 2, and supervising the fidelity of the set labels. Therefore, the training dataset has been customized for our specific task.

In Table 1, we show a summary of the datasets. As an indicator for the balance level of a given dataset, let us define Representation Balance. We propose the use of Shannon's Entropy for a Random Variable, considering the set of classes $\mathscr{U}$ and the probability distribution over the classes $P(u)$,

$$RB = -\frac{1}{\log_2(m)} \sum_{u \in \mathscr{U}} P(u) \log_2(P(u)), \tag{1}$$

where m is the number of classes in the dataset. The second factor of the equation is Shannon's Entropy, and $-\frac{1}{\log_2(m)}$ is a normalization factor. When RB is closer to 1 it means the the classes are more equiprobable. It is clear that *Affectnet* and *DAiSEE* sets are very unbalanced, a fact that will impact training and accuracy.

As two of our datasets contain videos, we must perform a preprocessing to extract the facial information from each frame and crop it to a size of 224x224. We use exhaustive frame extraction, which consists on using all frames in the video. This is done to obtain full detail over the sequences and configure the dataloader to get the samples configurable frame rate as a hyperparameter. By fine-tuning the frame rate, we expect to get a result indicating the optimum value, a trade-off between speed and accuracy.

Dataset	# Samples	# Subjects	Labels	Representation balance
Affectnet	1M images	450000	8 basic emotions	0.73
CK+	593 videos	123	7 basic emotions	0.91
DAiSEE	9068 videos	112	4 learning states	0.61

Table 1. Description of the datasets used for train and test

The baseline model was trained using a single *NVIDIA Titan X* for 40 epochs which took around 40 hours. The chosen batch size was 64 given by the limitations on GPU memory. The learning rate was empirically set at 0.0001 and we used Adam optimizer, also based on empirical results. The loss function used for training is Categorical Cross-Entropy, the standard for a classification problem.

A final note on training and data is about the problem of class balance. As shown with the RB indicator, table 1, some classes are over-represented as well as others that have fewer samples. This will generate biases in the predictions, where the less representative classes will not be well learned. We tackled this problem by using balance weights, which penalize highly imbalanced classes.

The trained model was developed using *Keras* [21] and *Tensorflow* [22] frameworks, and we employ *Tensorflow JS* in order to optimize the model for edge devices. The process of optimization removes all the layers and operations that are only used for training, maintaining just the operations needed for inference. Furthermore, the model is stored in a JSON file containing the inference operation and the trained parameters of the model.

3.4. Students Learning Reaction Dataset

This work in progress will follow up with the confection of the *Students Learning Reactions (SLR) Dataset*. The dataset will facilitate video recordings of students in a learning session and the video that the user was watching. The videos will be annotated using the GTF given by the user in significant instants, and also automatic annotations using the learning states predicted by the Deep Learning model. This dataset will allow researchers to attack many tasks in Computer Vision, such as Emotions Evoked in videos or Engagement Prediction, under an End User License Agreement and the approval of the Ethics committee at UPF (CIREP). For more information refer to Section 5.

4. Preliminary results

This section presents the trained baseline model results using train and validation sets depicted in table 2. Moreover, we describe experimental results obtained from recording a small set of participants.

4.1. Training Results

The precision score (in per cent unit) and categorical Cross-Entropy are used to judge the quality of each estimated affect labels with train and validation splits (80% - 20%).

Learning State	Val Loss	Val Acc.
Overall	**1.2257**	**0.8068**
Interested	0.6444	0.8418
Confused	1.2431	0.7534
Bored	0.6754	0.8754
Distracted	1.1654	0.6754

Table 2. Results of validation for each LS

Results for the baseline model using *MobileNetV2* are depicted in table 2. Overall validation metrics exhibit acceptable accuracy even with just involving space information. Looking at the breakdown over LS we can see drifts in the different classes. This due to the balance of the data. Results also prove the need to involve time-domain information to make the system more robust and precise.

4.2. Experimental test results

In this experiment, we present the system to many participants and let them perform a learning session. The goal is to test the functionalities of the system and also validate the performance of the prediction model, comparing the predictions with the given GTF. Finally, assess the learning process at the three levels, namely student, video and group.

Table 3 presents some metrics on inference speed and video quality for different device configuration. The difference in inference speed is big when comparing high-range with low-range computers, however the values are acceptable in terms of user experience. In some cases when the computer is being stressed, video quality can get penalized due to the processing of the images.

Configuration	Hardware	Inference speed	Video Quality
Mobile device	Android, 3GB RAM	297ms	Subtle delays on face detection
Basic laptop	Intel i5, 16GB RAM	280ms	Subtle delays on face detection
Mid range computer	Intel i7, 16GB RAM	162ms	No impact
High range computer	Intel i9, 32GB RAM	73ms	No impact

Table 3. Performance comparison for different hardware configurations.

From the sessions performed with 20 students, we compute an MVR over intervals of 1 minute and compare the predictions with the GTF given in important parts of the video. In this case, we ask to answer which learning state are they experimenting with the video. The final precision score is 57%, which is lower than the predictions from the validation set. Empirically, we see that works very good on some participants and poorly on some others. This is due to video quality, illumination conditions and facial expressions not seen in training.

One remarkable observation is that the MVR at group level, averaging the LS overall students, denotes where are the critical instants of the narrative structure. When there are abrupt changes in the narrative, visually or conceptually, the learning states also change.

This work does not provide a comparison with other methods, as it was trained in a transfer-learning manner, using a customized version of DAiSEE dataset. Also, we cannot compare results on speed, as we have no reported values of other alternatives.

5. Ethics, fairness and privacy

In this work, we are very concerned with the behaviour of Artificial Intelligence algorithms regarding fairness and the occurrence of unwanted biases in the predictions. Labels that are over/under-represented in the training data yield models that disadvantage minority groups and vice versa. We attack this problem by balancing training and validation data over classes. We also adopt the approach of accounting predictions that have confidence above 70%, and ignoring the predictions below that threshold.

All works that involve user data acquiring and processing must be assessed and certified by an Ethics Committee, even more if the data is bio-metric and invasive. In that sense, this work is being evaluated by CIREP organism at UPF.

The requirement for being able to participate in the experiments is the given consent for data treatment that includes video recording and processing of facial information, as well as the publication of the resulting dataset for research purposes. Students that contribute to the *SLR Dataset* will sign the policy that will give privacy rights assured in the Spanish RGPD, and research groups that may use this dataset will get access by means of signing an EULA.

It is notorious that using this type of data (Facial and Emotional) can be intrusive, so we must assure that Face Expression-based methods only contribute to the improvement of virtual experiences, such as educational, medical or security.

6. Conclusions

In this paper we proposed a framework to improve the E-learning experience, with mechanics for predicting learning states and getting student feedback during video visualization. The method is based on facial expression analysis and CNN. The system is able to link video time with learning state, and in the analysis the teacher can follow the evolution at three levels of detail, student, video and group. The proposed baseline architecture presents acceptable performance with small computational resources. However, future work must be focused on studying more sophisticated models, that involve the temporal dimension of the data. Through experiments, we foresee that the narrative of the video has a direct impact on the learning states of the students. Finally, the system is ready to collect data in order to create the Students Learning Reactions Dataset, to find better solutions using Computer Vision.

Acknowledgements We thank nanoMoocs project team and Ris3Cat Media Community for supporting this research. We also thank the IT team at UPF for technical support with web deployment and GPU usage.

References

[1] R. W. Picard. *Affective computing*. MIT press, 2000.

[2] J. Chen, N. Luo, Y. Liu, L. Liu, K. Zhang, and J. Kolodziej. A hybrid intelligence-aided approach to affect-sensitive e-learning. *Computing*, 98(1-2):215–233, 2016.

[3] J. A. Russell. A circumplex model of affect., 1980.

[4] L. Shen, M. Wang, and R. Shen. Affective e-learning: Using emotional data to improve learning in pervasive learning environment. *Journal of Educational Technology & Society*, 12(2):176–189, 2009.

[5] M. Porta, S. Ricotti, and C. J. Perez. Emotional e-learning through eye tracking. In *Proceedings of the 2012 IEEE Global Engineering Education Conference (EDUCON)*, pages 1–6. IEEE, 2012.

[6] TS Ashwin, J. Jose, G. Raghu, and G. R. M. Reddy. An e-learning system with multifacial emotion recognition using supervised machine learning. In *2015 IEEE seventh international conference on technology for education (T4E)*, pages 23–26. IEEE, 2015.

[7] N. Alyuz, E. Okur, E. Oktay, U. Genc, S. Aslan, S. E. Mete, Eb Arnrich, and A. A. Esme. Semi-supervised model personalization for improved detection of learner's emotional engagement. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, pages 100–107, 2016.

[8] Z. Luo, C. Jingying, W. Guangshuai, and L. Mengyi. A three-dimensional model of student interest during learning using multimodal fusion with natural sensing technology. *Interactive Learning Environments*, 0(0):1–14, 2020.

[9] A. Kaur, A. Mustafa, L. Mehta, and A. Dhall. Prediction and localization of student engagement in the wild. In *2018 Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–8. IEEE, 2018.

[10] O. El Hammoumi, F. Benmarrakchi, N. Ouherrou, J. El Kafi, and A. El Hore. Emotion recognition in e-learning systems. In *2018 6th International Conference on Multimedia Computing and Systems (ICMCS)*, pages 1–6. IEEE, 2018.

[11] S. K. Gupta, TS Ashwin, and R. M. R. Guddeti. Students affective content analysis in smart classroom environment using deep learning techniques. *Multimedia Tools and Applications*, 78(18):25321–25348, 2019.

[12] M. Megahed and A. Mohammed. Modeling adaptive E-Learning environment using facial expressions and fuzzy logic. *Expert Systems with Applications*, 157:113460, 2020.

[13] K. P. Rao and M V P C. K. Rao. Recognition of Learners ' Cognitive States using Facial Expressions in E-learning Environments. 22(12):93–103, 2020.

[14] A. Sun, Y. J. Li, Y. Min Huang, and Q. Li. Using facial expression to detect emotion in e-learning system: A deep learning method. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10676 LNCS:446–455, 2017.

[15] M. Mukhopadhyay, S. Pal, A. Nayyar, P. K. D. Pramanik, N. Dasgupta, and P. Choudhury. Facial emotion detection to assess learner's state of mind in an online learning system. In *Proceedings of the 2020 5th International Conference on Intelligent Information Technology*, pages 107–115, 2020.

[16] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen. MobileNetV2: Inverted Residuals and Linear Bottlenecks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 4510–4520, 2018.

[17] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager. Temporal convolutional networks for action segmentation and detection. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017-January:1003–1012, 2017.

[18] P. Lucey, J. F Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *2010 ieee computer society conference on computer vision and pattern recognition-workshops*, pages 94–101. IEEE, 2010.

[19] A. Mollahosseini, B. Hasani, and M. H. Mahoor. AffectNet: A database for facial expression, valence, and arousal computing in the wild. *arXiv*, 2017.

[20] A. Gupta, A. D'Cunha, K. Awasthi, and V. Balasubramanian. DAiSEE: Towards User Engagement Recognition in the Wild. 14(8):1–12, 2016.

[21] F. Chollet et al. Keras. https://keras.io, 2015.

[22] M. Abadi, P. Barham, K. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283, 2016.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210156

No-Reference Digital Image Quality Assessment Based on Structure Similarity

Basma AHMED[1], Mohamed ABDEL-NASSER[2,3], Osama A. OMER[3], Amal RASHED[1] and Domenec Puig[2]
[1]Faculty of Computers and Information, South Valley University, Qena, Egypt
[2]Computer Engineering and Mathematics Department, University Rovira i Virgili, Tarragona, Spain
[3]Electrical Engineering Department, Aswan University, Aswan, Egypt

Abstract. Blind or non-referential image quality assessment (NR-IQA) indicates the problem of evaluating the visual quality of an image without any reference, Therefore, the need to develop a new measure that does not depend on the reference pristine image. This paper presents a NR-IQA method based on restoration scheme and a structural similarity index measure (SSIM). Specifically, we use blind restoration schemes for blurred images by reblurring the blurred image and then we use it as a reference image. Finally, we use the SSIM as a full reference metric. The experiments performed on standard test images as well as medical images. The results demonstrated that our results using a structural similarity index measure are better than other methods such as spectral kurtosis-based method.

Keywords. Blind image quality assessment (BIQA), deblurring, point spread function (PSF), structural similarity index measure (SSIM).

1. Introduction

Image quality assessment (IQA) is important for numerous image processing measures, such as re-extraction, restoration, enhancement, compression, and acquisition [1]. IQA is divided into Non-reference NR IQA that's refer to the automatic evaluation of image quality using an algorithm so that the only information the algorithm receives before it predicts quality is the distorted image whose quality is obtained, Full-reference FR IQA that requires as input not only the distorted image, but also pristine image which the distorted image quality is evaluated, reduced reference (RR) approaches that possess some information regarding the reference image, but not the actual reference image itself, regardless of the distorted image [11].

Images are affected by different types of distortion during transmission and processing. Therefore, their quality must be addressed or evaluated before they are used. Image quality evaluation is used in various applications such as enhancement, recovery, compression, acquisition, etc. Motion blur is one of the most common artifacts in digital photography [2] [3]. When taking a photo in dim light with a portable camera, tilting the photographer's hand to shake can blur the image. In response to this problem, image deblurring has become an active topic in computational photography and image processing in recent years [6].

The images are blurred due to many reasons such as defects in capturing pictures, low intensity during camera exposure, atmospheric problems, lens defocus, etc. Human visual systems are good at being aware of it. But the mechanism of this processing is not fully understood. Therefore, it is difficult to come up with metrics to estimate blur in images. In digital images, there are 3 types of blur effects: average blur, gaussian blur, motion blur. Recently, efforts have been made to develop such blind IQMs devices including spectral kurtosis [4] [13], Blind Referenceless Image Spatial Quality Evaluator (BRISQUE) [16], and the Natural Image Quality Evaluator (NIQE) [17], etc. Some of these IQMs rely on the statistical properties of the deblurred image while others rely on measuring the image quality by basing them on the Human Visual System (HVS)[14][15][20]. Min et al.[6] proposed a blind image quality assessment based on a pseudo reference image and they are using "reference" called pseudo reference image (PRI), and a PRI based blind IQA (BIQA) framework, they develop specific measures of distortion to estimate blockiness, sharpness, and noisiness. PRI-based scales calculate the similarity between distorted image structures and PRI structures. Through a two-stage quality regression after the distortion identification framework, they then incorporate metrics for PRI-based distortion into a general-purpose BIQA method called the PRI-based blind scale (BPRI). Moreover, BPRI not only performs well in landscape images, but also applies to screen content images. Moorthy and Bovik [15] proposed a blind image quality assessment (IQA) based on the hypothesis that landscapes possess certain statistical properties that are altered in the presence of distortion, rendering them unnatural; and that by characterizing this anomaly using Scene Statistics. Accordingly, they presented an (NR)/blind Algorithm, the distortion identification-based image verity and integrity evaluation (DIIVINE) index, that evaluates the quality of the distorted image without the need for a pristine image. DIIVINE is based on a two-stage framework that includes distortion identification followed by assessing the quality of the specified distortion. DIIVINE can evaluate the quality of a distorted image across multiple distortion classes, against most NR IQA algorithms of a distortion-specific nature.

Lin et al. [19] suggested a simple but highly effective full reference IQA method using Visual Saliency (VS). In the suggested image quality assessment (IQA) model, the role of VS is divided into two parts. The first part, Visual Saliency is used as a feature when computing the local quality map of the distorted image. In the second part, when aggregating the Quality Score, VS is used as a weighting function to reflect the local area's importance. The proposed IQA index is called a visual saliency-based index (VSI). The proposed IQA index VSI performs better while maintaining a moderate computational complexity.

Yue and Jiangming [14] suggested a method for deblurring image method based on local edge selection. The local edges are determined by the difference between the bright channel and the dark channel, after that, a new image deblurring model is created including the term Local Edge Alignment. Obtaining a clear image and core blurring is based on alternating iterations, where the clear image is obtained by ADMM.

Arthur and Lionel [8] proposed a deeper look at three recent measures of severity (global phase coherence, sharpness index, and a simplified version of it), All of which a probabilistic sense measure the surprisingly small overall contrast of the image compared to that of some of the associated random phase fields. They display many theoretical connections between these indicators and study their behavior in a general class of fixed random fields. In the end, they suggested an application to idiosyncratic blindness and demonstrated its efficiency in several examples.

This paper examines the efficiency of some existing blind IQMs for blind image deblurring like spectral kurtosis and compares it with the proposed SSIM metric as a blind image quality measures IQM [9].

2. Methodology

Image degradation model can be described by:

$$g = H * f + n \tag{1}$$

where g is the blurry image, H indicates the distortion facto, known as the point spread function (PSF), n is added noise, and * represents the convolution operator [8]. In the spatial field, the PSF characterizes the rate at which the optical system blurs the spotlight [1][10], Figure1 illustrates image deblurring model.

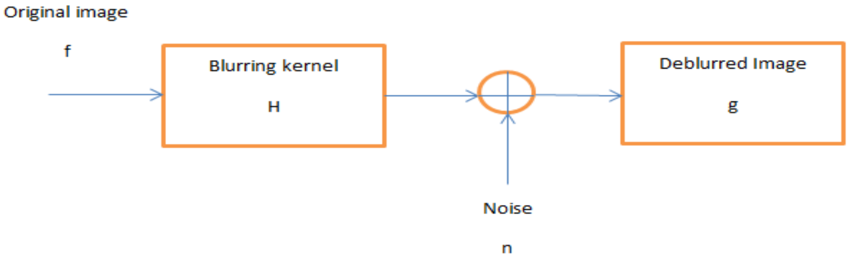


Figure 1. Deblurring image.

In the proposed algorithm a blurred image is obtained by convolving the original image and blurring kernel (PSF). The PSF parameters (angle and length) can be calculated by first estimating the angle quite accurately using analysis in the Cepstrum domain [12], for a given angle we can estimate the length of the blur in the image for a given angle [1]. The Winner filter is applied to the blurred image to get the true image as the equation described below:

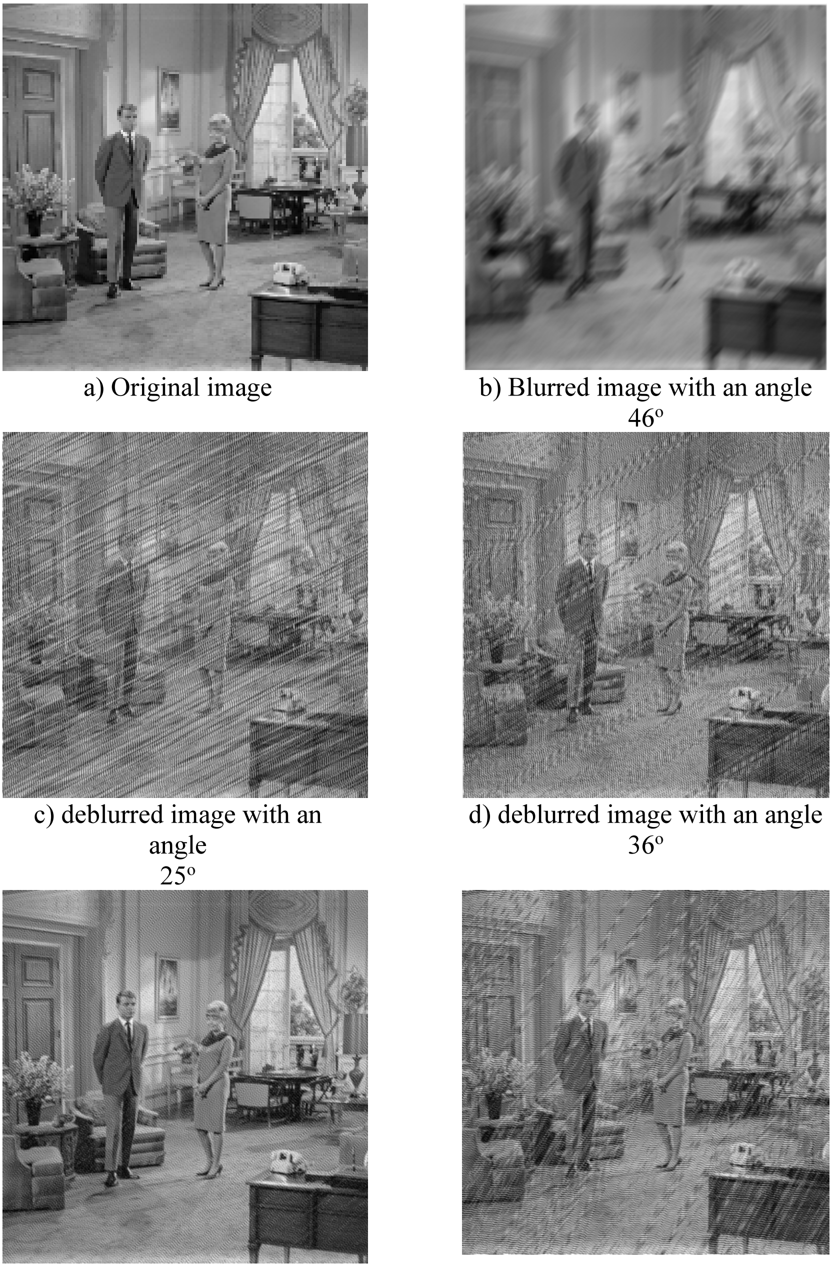
$$G(u,v) = \frac{H^*(m,n)}{|H(m,n)|^2 + NSR} \tag{2}$$

where H is the blurring filter and NSR is the noise variance. When the filtered PSF is similar to the real PSF (h) it will be able to reproduce the same blur in reblurred image the restoration filter will produce less noise and resonance. SSIM (structural similarity index measure) is measured for blurred [18], reblurred image to emphasize similar blur reproduction [5][6]. The next pseudocode presents the steps of the proposed reblurring algorithm.

Algorithm1: Pseudocode for the proposed reblurring algorithm.

```
1: Input pristine image
2: Determine the blurred image by convolving the original image and blurring kernel (PSF) using equation (1) The PSF parameters (angle and length) can be calculated using analysis in the Cepstrum domain
```

3: Determine the deblurring image using Winner filter as equation (2)
4: The image has been retrieved by a candidate blurring kernel (PSF)
5: Using SSIM to measure the similarity between blurred and reblurred images.



a) Original image

b) Blurred image with an angle 46^{o}

c) deblurred image with an angle 25^{o}

d) deblurred image with an angle 36^{o}

e) deblurred image with an angle 46^{o}

f) deblurred image with an angle 57^{o}

Figure 2. Examples for blind deblurring schema

An example to illustrate a blind deblurring method is shown in Figure 2(a) the original livingroom image. Figure 2(b) shown livingroom image which was blurred with motion PSF at an angle of 46 degrees. Figure 2(c) and (d)(e)(f) show the Deblurred images due to PSF angle 25 and, 36,46, 57 degrees respectively. Figure 2(e) shows The deblurred image is similar to the original image and the noise level and ringing are tolerated. The image deblurred with a blur kernel similar to the true PSF shown in is the one that will reproduce a similar blur when reblurred.

3. Experimental Results and Discussion

Deblurring results using the scheme for images are presented, these include images under motion blur. Experiments for The proposed algorithm include testing of "Standard" test images (a set of images found frequently in the literature: Lena, peppers, cameraman, pirate, etc., all in uncompressed tif format and of the same 512 x 512 size, and some medical images (Eye Fundus image, CT image, etc) as shown in Figure 3.

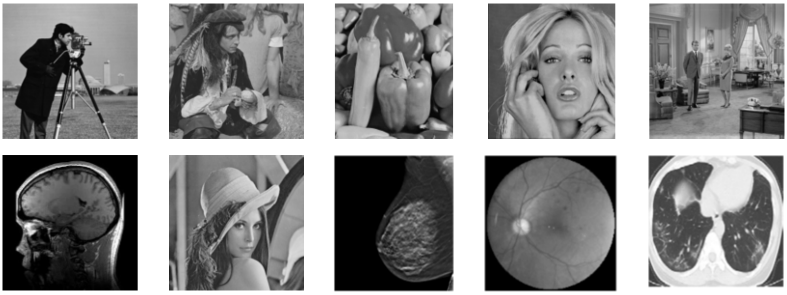


Figure 3. Example of test images

Table 1 summarizes the SSIM results for the blurred. In this case, motion blurred images were used and the PSF parameters angle, theta, was estimated using Cepstrum analysis. From the results shown in Table 1, we find the estimated values close to the theta values used to blur images. SSIM has been calculated for three sets of images: blurred and reblurred, original and blurred, and original and deblurred. The original blurred image and the reblurred image are compared using SSIM and shown by the blurred-reblurred pair column in Table 1. The high values of SSIM depict the reblurred images are in a close approximation of the original blurred images. Therefore the reblurring can estimate the blurring PSFs in the case of motion-blurred images. A higher value of SSIM shows an image of high quality.

In this paper, SSIM were scrutinized. The reblurred images is obtained from re-convolved the deblurred images with the estimated PSF. It turns out that if the candidate PSF is close to the original PSF, it will produce the same blurring in the reblurred image as the original blurred image. As it is evident SSIM based deblurred images produced better visual quality, in the case of motion deblurring.

Image	Length using Cepstrum analysis	Original Angle	Estimated Angle	SSIM		
				Blurred Reblurred	Original Blurred	Original Deblurred
livingroom	23	46	46	0.8936	0.4759	0.5187
pirate	21	44	44.1	0.8572	0.4836	0.5441
woman_blonde	14	23	23.2	0.9064	0.6113	0.6474
Mandrill	31	46	46.5	0.9344	0.7948	0.8104
Camera man	17	33	33.2	0.8456	0.5114	0.5571
Goldhill	23	27	26.9	0.8716	0.4177	0.4733
Lena	25	13	13	0.9130	0.4747	0.5303
Peppers	20	42	41.9	0.8442	0.4004	0.5921
Eye Fundus Image	31	41	40.3	0.9525	0.8887	0.8964
Tomosynthesis	23	26	26.1	0.9743	0.8658	0.8677
CT image	11	32	32.2	0.7826	0.4298	0.4910
MRI	13	45	45.8	0.7050	0.5105	0.5781

Table 1. SSIM results for the test images

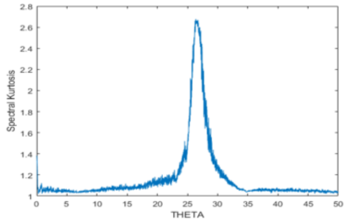
4. A comparison between the proposed algorithm and spectral kurtosis for image quality assessment.

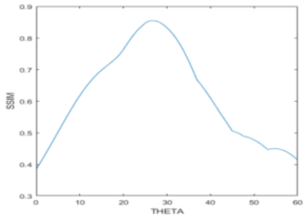
By applying the comparison between the proposed scheme and spectral kurtosis method on some images such as (a) corn, (b) boat, (c) barbara, (d) woman darkhair as in Figure 4 with blur angle of 26.5,29, 32.35,47 degrees respectively. Our proposed scheme SSIM error measure estimated the angle as 26.40, 29.1, 32.5, 46.8degrees respectively while spectral kurtosis estimated 26.38,29.3,31.9,49.6 as the blur angle. The corresponding SSIM, spectral kurtosis plot for deblurring images is also shown in Figure 5. We deduce from the shown graph that the SSIM measure maximizes the near the perimeter of the true blurring parameter value. This confirms that our results using a structural similarity index measure are better than spectral kurtosis method.

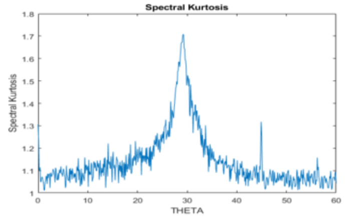
(a) **(b)**

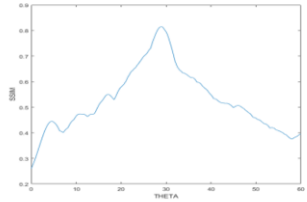
(c) **(d)**

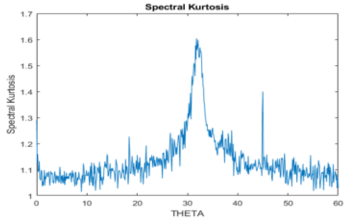
Figure 4. Examples of images(a)-(d)

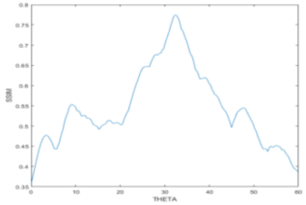
<table>
<tr><th rowspan="2">Image</th><th colspan="2">Algorithm</th></tr>
<tr><th>Spectral kurtosis</th><th>The proposed algorithm(SSIM)</th></tr>
<tr><td>Corn</td><td>


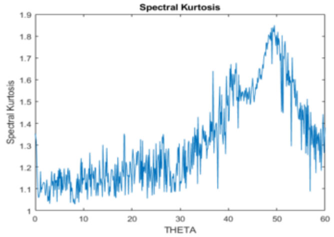
</td><td>


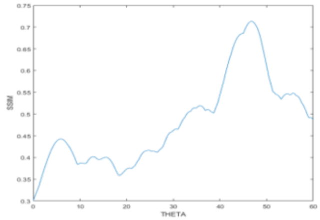
</td></tr>
<tr><td>Boat</td><td>


</td><td>


</td></tr>
<tr><td>Barbara</td><td>


</td><td>


</td></tr>
<tr><td>Woman
darkhair</td><td>


</td><td>


</td></tr>
</table>

Figure 5. The corresponding SSIM, spectral kurtosis plot for deblurring images

Conclusions

In this paper, a full-reference blind quality measure SSIM was proposed as a deblurring measure. The proposed SSIM performs well in the case of deblurring by motion-blurred images. The proposed SSIM has been rigorously tested and compared with the other state-of-the-art IQA indices like spectral kurtosis on different images such as Standard" test images and some medical images. The results demonstrated that the proposed IQA index SSIM could yield much better-quality deblurring images as compared to others measures such as spectral kurtosis. In future work, for Blind Image Quality Measures (IQMs), it may be interesting to using other metrics and developing them.

Acknowledgment

The Spanish Government partly supported this research through Project PID2019-105789RBI00.

References

[1] Chang C-F, Wu J-L, Tsai T-Y. A single image deblurring algorithm for nonuniform motion blur using uniform defocus map estimation. Mathematical Problems in Engineering. 2017.

[2] Almeida MS, Figueiredo MA. Parameter estimation for blind and non-blind deblurring using residual whiteness measures. IEEE Transactions on Image Processing. 2013;22(7):2751-63.

[3] Ahmed IT, Der CS, Hammad BT. RECENT APPROACHES ON NO-REFERENCE IMAGE QUALITY ASSESSMENT FOR CONTRAST DISTORTION IMAGES WITH MULTISCALE GEOMETRIC ANALYSIS TRANSFORMS: A SURVEY. Journal of Theoretical & Applied Information Technology. 2017;95(3).

[4] Smith WA, Fan Z, Peng Z, Li H, Randall RB. Optimised Spectral Kurtosis for bearing diagnostics under electromagnetic interference. Mechanical Systems and Signal Processing. 2016;75:371-94.

[5] Khan A, Yin H. Efficient blind image deconvolution using spectral non-Gaussianity. Integrated Computer-Aided Engineering. 2012;19(4):331-40.

[6] Min X, Gu K, Zhai G, Liu J, Yang X, Chen CW. Blind quality assessment based on pseudo-reference image. IEEE Transactions on Multimedia. 2017;20(8):2049-62.

[7] Oszust M. Local feature descriptor and derivative filters for blind image quality assessment. IEEE Signal Processing Letters. 2019;26(2):322-6.

[8] Leclaire A, Moisan L. No-reference image quality assessment and blind deblurring with sharpness metrics exploiting fourier phase information. Journal of Mathematical Imaging and Vision. 2015;52(1):145-72.

[9] Ding K, Ma K, Wang S, Simoncelli EP. Image quality assessment: Unifying structure and texture similarity. arXiv preprint arXiv:200407728. 2020.

[10] Muthana R, Alshareefi AN, editors. Techniques in De-Blurring Image. Journal of Physics: Conference Series; 2020: IOP Publishing.

[11] Jaramillo Sr BO, Niño-Castañeda JO, Platiša L, Philips W. Content-aware objective video quality assessment. Journal of Electronic Imaging. 2016;25(1):013011.

[12] Kumar A. Deblurring of motion blurred images using histogram of oriented gradients and geometric moments. Signal Processing: Image Communication. 2017;55:55-65.

[13] Kang J, Zhang X, Teng H, Zhao J. Application of maximum correlated Kurtosis deconvolution on bearing fault detection and degradation analysis. Vibroengineering PROCEDIA. 2014;4:119-24.

[14] Han Y, Kan J. Blind Image Deblurring Based on Local Edges Selection. Applied Sciences. 2019;9(16):3274.

[15] Moorthy AK, Bovik AC. Blind image quality assessment: From natural scene statistics to perceptual quality. IEEE transactions on Image Processing. 2011;20(12):3350-64.

[16] Mittal A, Moorthy AK, Bovik AC. No-reference image quality assessment in the spatial domain. IEEE Transactions on image processing. 2012;21(12):4695-708.

[17] Zhang L, Zhang L, Bovik AC. A feature-enriched completely blind image quality evaluator. IEEE Transactions on Image Processing. 2015;24(8):2579-91.

[18] Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing. 2004;13(4):600-12.

[19] Zhang L, Shen Y, Li H. VSI: A visual saliency-induced index for perceptual image quality assessment. IEEE Transactions on Image processing. 2014;23(10):4270-81.

[20] Li C, Guan T, Zheng Y, Zhong X, Wu X, Bovik A. Blind image quality assessment in the contourlet domain. Signal Processing: Image Communication. 2021;91:116064.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210157

Reliable Deep Learning Plant Leaf Disease Classification Based on Light-Chroma Separated Branches

Joao Paulo SCHWARZ SCHULER [a], Santiago ROMANI [a]
Mohamed ABDEL-NASSER [a] Hatem RASHWAN [a] and Domenec PUIG [a]
[a] *Universitat Rovira i Virgili*

Abstract. The Food and Agriculture Organization (FAO) estimated that plant diseases cost the world economy $220 billion in 2019. In this paper, we propose a lightweight Deep Convolutional Neural Network (DCNN) for automatic and reliable plant leaf diseases classification. The proposed method starts by converting input images of plant leaves from RGB to CIE LAB coordinates. Then, L and AB channels go into separate branches along with the first three layers of a modified Inception V3 architecture. This approach saves from 1/3 to 1/2 of the parameters in the separated branches. It also provides better classification reliability when perturbing the original RGB images with several types of noise (salt and pepper, blurring, motion blurring and occlusions). These types of noise simulate common image variability found in the natural environment. We hypothesize that the filters in the AB branch provide better resistance to these types of variability due to their relatively low frequency in the image-space domain.

Keywords. DCNN, CNN, Plant Leaf Disease, Classification, Computer Vision, Plant Village, Deep Learning

1. Introduction

Plant leaf images taken in the field and away from controlled laboratory conditions frequently suffer from blurring, motion blurring, occlusion and illumination variations. Automated detection systems frequently suffer from these common adverse effects. Inspired on Multi-path Convolutional Neural Networks [1] and Dual Paths Neural Networks [2], we created an Inception V3 [3] based architecture that has two branches (paths) along the first 3 convolutional layers. One branch is fed with the achromatic L channel, while the other branch is fed with AB channels provided by the input CIE Lab color coordinate space. In this work, we study 3 two-branches Inception V3 variants: 20%L-80%AB, 50%L-50%AB and 80%L-20%AB. In this notation, the percentages indicate the proportion of the original number of neurons of each separated layer dedicated to each path. This two-branches solution provides more resistance to adverse effects such as blurring. For this work, we are training our architecture with the PlantVillage dataset [4] that contains classes for 12 healthy crops and 26 crop diseases.

This article is structured as follows: section 2 presents and discusses relevant work in regards to computer vision, DCNNs and image based plant disease diagnostic. Section 3 presents the proposed method. The results and the discussion are given in sections 4 and 5. Section 6 summarizes the main conclusions.

2. Related work

In a previous work [5], training a CNN with input images encoded in the CIE Lab color space, we were able to show that we can classify the CIFAR-10 dataset [6] more efficiently and with higher classification accuracy by creating an architecture that has a subpath dedicated to light and another subpath dedicated to color channels. In this previous work, each subpath has only the first convolution layer dedicated for each L and AB channels.

A number of machine learning methods have been proposed specifically for image based plant disease diagnostic [7,8]. Mohanty et. al. [9] worked with AlexNet and GoogLeNet models for the PlantVillage dataset classification. They trained both models from scratch and with transfer learning. They also experimented feeding their models with RGB and grayscale images. They found better results feeding RGB images to both tested models. Their best result without transfer learning was 98.37%. Geetharamani et al. [10] classified the PlantVillage dataset with 3 convolutional, 2 max poolings and 2 dense layers achieving 96.46% of accuracy. Toda at al. [11] working with a trimmed Inception V3 showed that DCNNs can learn the colors and textures specific to plant leaf diseases resembling human made classification.

3. Methodology

Figure 1 shows two designs of CNNs for plant disease classification. Toda & Okura's [11] proposed an Inception v3 variation that gets rid of the last 5 mixed layers (out of 11). The authors proved that it is enough for the sake of classification PlantVillage dataset. Therefore, we have chosen their model as our baseline.

The design shown on the right of figure 1 corresponds to our proposal, which splits the first three convolution layers of the baseline model into two branches, one for the L channel and the other for the AB channels from the transformed RGB image. Then, the output from each branch is concatenated and the rest of the network is the same as the baseline.

Another relevant remark is that we use a hyperparameter that determines the distribution of a fixed number of filters among L and AB branches, which allows us to look for the optimal contribution of each branch to the classification task. This distribution is implemented with the value of a variable x, shown in figure 1 as the number of L filters in the third layer. In the original Inception V3 implementation, the first three convolutional layers have 32, 32 and 64 filters, respectively. We have analyzed three configurations of the two-branch design named after the percentage of filters dedicated to L and AB branches: 20%L-80%AB, 50%L-50%AB and 80%L-20%AB. The resulting number of filters for each variant is shown in the table 1.

Since we intend to compare our variants with the baseline as fairly as possible, the sum of filters of the two branches in each layer is the same as in the Inception V3 design.

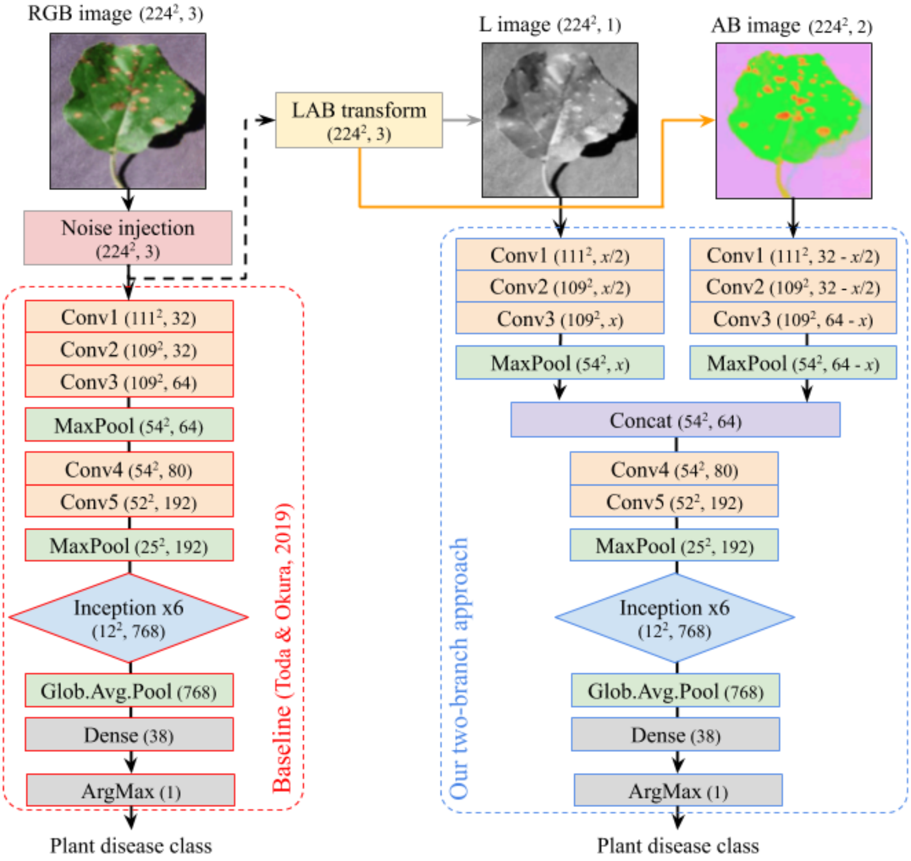


Figure 1. Graphical representation of the worked network architectures: on the left, the Toda & Okura's single-branch (baseline) approach fed with an RGB image; on the right, our two-branch approach fed with L+AB images. The x expressions determine a varying number of filters in L branch and a complementary number of filters in AB branch.

Model	1st & 2nd Layers	3rd Layer
baseline	32	64
20%L + 80%AB	6 — 26	13 — 51
50%L + 50%AB	16 — 16	32 — 32
80%L + 20%AB	26 — 6	51 — 13

Table 1. Number of filters in 1st, 2nd and 3rd layers of the baseline and our variants. For our variants, we have the number of filters in the L branch at the left and in the AB branch at the right.

However, our design saves from 1/3 to 1/2 of weights and computational floating point operations in the split layers, as shown in tables 2. Despite the reduction in weights, the learning capacity of our models is not degraded since our three variants achieve similar accuracy (99.48%, 99.11%, 99.08%) to the one provided by the baseline (99.32%).

Our design is based on the well-known fact that RGB channels are highly correlated among each other [12] in the sense that shading and shadows render a set of different RGB values from the intrinsic color(s) of a surface. Specifically, intensity variations induced by illumination variation, edges and texture modify the three RGB values at same proportion. Hence, transforming RGB channels into some sort of achromatic-chromatic space, like CIE Lab, effectively isolates the gray-level features in the L channel and the color-related features in the AB channels. We are forcing the filters in each branch to learn features related to the nature of each cue, i.e., we expect that L filters will focus on texture and edges of the leafs (intrinsic shape, damaged leaf areas, etc.) while the AB filters will focus on color findings (lesions, general color of the leaf, etc.).

model	weights (Saving)	flops (Saving)
baseline	28512	701M
20%L + 80%AB	19746 (31%)	485M (31%)
50%L + 50%AB	14256 (50%)	350M (50%)
80%L + 20%AB	19566 (31%)	481M (31%)

Table 2. Weights and required forward pass floating point operations along the first 3 convolutional layers in baseline and our variants.

To verify the reliability of the baseline and our variants, we have included one module for noise injection. This allows us to perturb the original RGB images with different types of artifacts and varying degrees of severity of those artifacts. It must be observed that the noise injection is previous to the RGB-to-LAB transformation.

Our code was coded with Keras/Tensorflow v2.2. We rented cloud based hardware with NVIDIA GPUs, intel CPUs and virtual machines from 32GB to 64GB of RAM. The implementation details of our approach are strongly based on the reference paper [11]. Each convolutional layer is composed of a 2D convolution, a batch normalization and a ReLU activation function. All convolutional filters from Conv1 to Conv5 are of the size 3×3 except for Conv4 which is 1×1. The optimization method is stochastic gradient descent, and the loss function is weighted categorical cross entropy to compensate for unbalanced number of samples among classes. The batch size is 32 and we store the weights that obtain the best validation accuracy in 30 epochs. We trained all models from scratch. The noise injection module has not been used for training since this module is only intended to verify the reliability of the models under controlled perturbation of the test images.

4. Results

Figure 2 shows the evolution of test accuracy in the studied models, baseline, two-branch 20%L-80%AB, 50%L-50%AB and 80%L-20%AB, for different types of noise and a range of noise amount.

In Salt and Pepper experiments, the range of noise indicates the percentage of pixels of the input image that have been changed to either white or black pixels (see Fig. 3 for an example). This type of noise simulates spuriously saturated values in the input signal. The corresponding plot depicts the 20%L-80%AB variant as the most reliable when the percentage of noisy pixels is above 3%. Above 3%, the classification accuracy is up to 10% more accurate than the baseline. Nevertheless, the baseline holds better performance than the other two branched models in the range of noise used for these experiments.

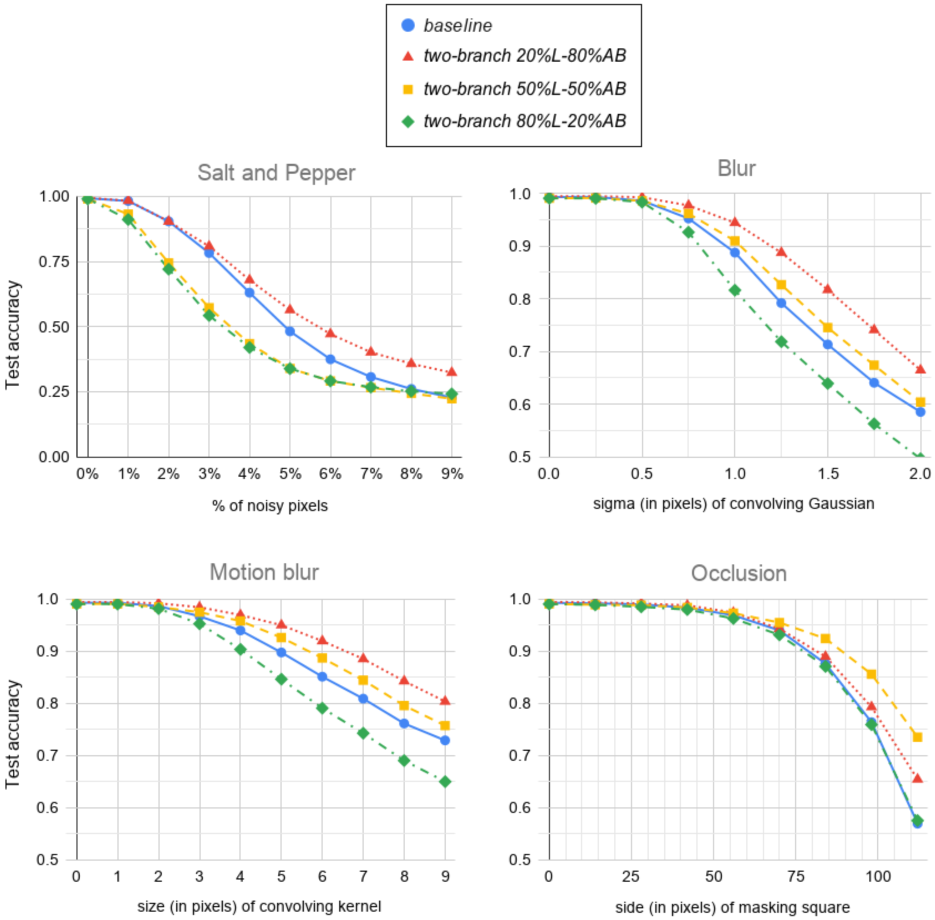


Figure 2. Result plots showing the test accuracy evolution of four approaches under a range of perturbation with four types of noise.

In Blur experiments, a Gaussian distribution of a given sigma in image space coordinates (distance in pixels) is convolved with the input RGB image values producing the typical blurring effect (check Fig. 3). This type of noise simulates unfocused snapshots or dirty lenses. In the corresponding plot, our 20%L-80%AB variant proves the most reliable under the tested range of sigmas. From $\sigma = 1.25$ to $\sigma = 1.75$, this best model overcomes the baseline by 10% of test accuracy. Moreover, the 50%L-50%AB variant also overcomes the baseline, although by a slight difference.

Motion blur is similar to blur (also check Fig. 3), but instead of a Gaussian distribution we use a sparse matrix of a given size with all cells equal to zero except for one line of cells, which is filled with ones divided by the number of cells in that line. By convolving the image pixel values with such a matrix (kernel), it is possible to simulate the blurring due to sudden camera shifts. The direction of movement is parallel to the line of cells different to zero. The extend of movement is equivalent to the length of that line. The corresponding plot depicts similar behavior to the blurring plot, although it is

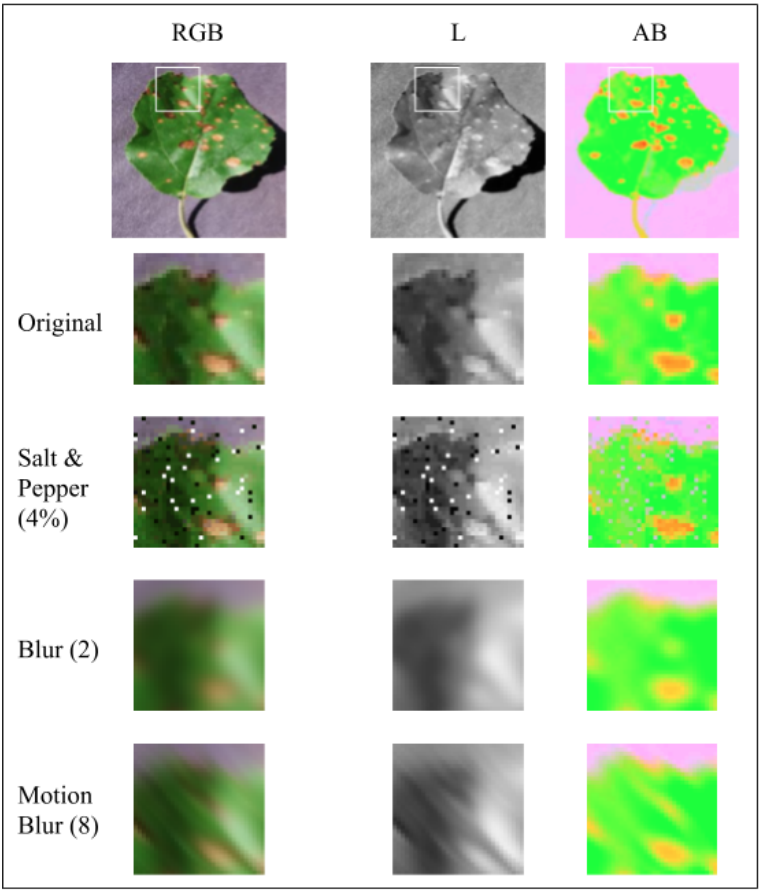


Figure 3. Noise injection in a portion of a test image (Apple Black Rot num.5), in RGB, L and AB spaces: Salt & Pepper noise in 4% of the image pixels; Blur by convolving a Gaussian bell with $\sigma = 2$ pixels; Motion Blur in up-left direction with 8 pixels of kernel width.

necessary to use a 9 pixels-side kernel to degrade the test accuracy of the 20%L-80%AB variant as much as with a $\sigma = 1.5$ in the blurring experiment.

Occlusion is performed by overlapping a square of gray pixels of a given size in a random position of the image. This type of noise simulates the occlusion of the target leaf by other non-interesting objects such as tree branches, fruits, etc. For these experiments, the model that renders the best reliability in the corresponding plot is our 50%L-50%AB variant with a remarkable difference of 5% above the second best model, the 20%L-80%AB variant, which in turn is also 5% above baseline and the 80%L-20%AB variant when the side of the masking square is beyond 100 pixels.

5. Digression

All results are highly determined by the fact that the leaf shape and their lesions are less varying in AB channels than in RGB and L channels as can be seen in the example of

Figure 3. In other words, the leaf representation in AB channels render broad areas of similar colors. This low-frequency nature of the AB channels makes the color-trained filters to inherently take into account a wider field of view. Therefore, more erroneous pixels are needed to mislead the classification. In contrast, the same leaf surface renders more frequent variations in RGB channels which provokes that their trained filters will have a smaller field of view. Specifically, high-frequency noise affects more to gray-level filters, which are actually the ones projected into the L channel. These observations may explain why focusing 80% of the filters on the AB branch provides the best results in presence of most types of noise.

For salt and pepper noise, the effect of spurious pixels in AB channels is noticeable but the larger field of view of corresponding filters allows to overcome those perturbed values. In the other hand, the field of view of L and RGB filters is closer to the area of each erroneous pixel. However, the baseline is more reliable than 50%L-50%AB and 80%L-20%AB configurations because its filters can better treat the spurious changes in the 3D RGB space than the combination of the split L and AB filters.

In contrast to salt and pepper, blurring is a perturbation of low-frequency nature. Despite this fundamental difference, our 20%L-80%AB configuration becomes again the most reliable. In this case, the smoothing of pixel values degrades more the features encoded in the L and RGB channels than the features encoded in the AB channels. The 50%L-50%AB configuration is also stronger than the baseline. In regards to motion blurring, the 20%L-80%AB and 50%L-50%AB configurations are again the most reliable.

For the occlusions experiment, the 50%L-50%AB and 20%L-80%AB variants are the most resilient specially for mask sizes above 1/4 of the total image area. Again, the reasoning for this effect is that a big occlusion in the AB image removes less relevant details than the same occlusion in L and RGB images as the key features in AB channels are wider in image space than in L or RGB channels.

6. Conclusion

In this paper, we have suggested a two-branch CNN for plant disease classification where the first three convolutional layers are specialized in learning chromatic and achromatic features from the CIE Lab color space. Besides classifying original RGB images with similar accuracy and less weights, our experiments also show that our 20%L-80%AB and 50%L-50%AB models better classify input images under salt and pepper, blurring, motion blurring and occlusion by margins up to 10%.

With regards to the optimal distribution of filters among achromatic and chromatic branches, our experiments show that about 80% of the filters should go into the chromatic branch in order to provide the maximum reliability in front of different sources of noise. The reason behind this conclusion is based in the fact that color filters have wider field of view than lightness or RGB filters. Another reason is the color cue portrays highly relevant features for plant disease classification.

As we have Toda & Okura's Inception V3 based work as our baseline, we did our experiments with a modified Inception V3. It would make sense as a future work to try the same two-branches approach with an Inception V4 [13] model.

References

[1] Wang M. Multi-path Convolutional Neural Networks for Complex Image Classification. CoRR. 2015;abs/1506.04701. Available from: http://arxiv.org/abs/1506.04701.

[2] Chen Y, Li J, Xiao H, Jin X, Yan S, Feng J. Dual Path Networks. CoRR. 2017;abs/1707.01629. Available from: http://arxiv.org/abs/1707.01629.

[3] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the Inception Architecture for Computer Vision. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition; 2016. Available from: http://arxiv.org/abs/1512.00567.

[4] Hughes DP, Salath'e M. An open access repository of images on plant health to enable the development of mobile disease diagnostics through machine learning and crowdsourcing. CoRR. 2015;abs/1511.08060. Available from: http://arxiv.org/abs/1511.08060.

[5] Schler JPS. Optimizing CNNs first layer with respect to color encoding. In: Valls CJA, editor. 6th URV Doctoral Workshop in Computer Science and Mathematics. vol. 1. Universitat Rovira i Virgil. Tarragona, Catalunya, Spain: Universitat Rovira i Virgil; 2020. p. 4.

[6] Krizhevsky A. Learning multiple layers of features from tiny images; 2009.

[7] Ferentinos KP. Deep learning models for plant disease detection and diagnosis. Computers and Electronics in Agriculture. 2018;145:311 318. Available from: http://www.sciencedirect.com/science/article/pii/S0168169917311742.

[8] Sladojevic S, Arsenovic M, Anderla A, Culibrk D, Stefanovic D. Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification. Computational Intelligence and Neuroscience. 2016 Jun;2016:3289801. Available from: https://doi.org/10.1155/2016/3289801.

[9] Mohanty SP, Hughes DP, Salath M. Using Deep Learning for Image-Based Plant Disease Detection. Frontiers in Plant Science. 2016;7:1419. Available from: https://www.frontiersin.org/article/10.3389/fpls.2016.01419.

[10] G G, J AP. Identification of plant leaf diseases using a nine-layer deep convolutional neural network. Computers & Electrical Engineering. 2019;76:323 338. Available from: http://www.sciencedirect.com/science/article/pii/S0045790619300023.

[11] Toda Y, Okura F. How Convolutional Neural Networks Diagnose Plant Disease. Plant Phenomics. 2019 03;2019.

[12] Pouli T, Reinhard E, Cunningham DW. Image Statistics in Visual Computing. 1st ed. USA: A. K. Peters, Ltd.; 2013.

[13] Szegedy C, Ioffe S, Vanhoucke V, Alemi A. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning; 2017. Available from: https://www.aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14806.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210158

Grouped Pointwise Convolutions Significantly Reduces Parameters in EfficientNet

Joao Paulo SCHWARZ SCHULER [a], Santiago ROMANI [a]
Mohamed ABDEL-NASSER [a] Hatem RASHWAN [a] and Domenec PUIG [a]
[a] *Universitat Rovira i Virgili*

Abstract. EfficientNet is a recent Deep Convolutional Neural Network (DCNN) architecture intended to be proportionally extendible in depth, width and resolution. Through its variants, it can achieve state of the art accuracy on the ImageNet classification task as well as on other classical challenges. Although its name refers to its efficiency with respect to the ratio between outcome (accuracy) and needed resources (number of parameters, flops), we are studying a method to reduce the original number of trainable parameters by more than 84% while keeping a very similar degree of accuracy. Our proposal is to improve the pointwise (1x1) convolutions, whose number of parameters rapidly grows due to the multiplication of the number of filters by the number of input channels that come from the previous layer. Basically, our tweak consists in grouping filters into parallel branches, where each branch processes a fraction of the input channels. However, by doing so, the learning capability of the DCNN is degraded. To avoid this effect, we suggest interleaving the output of filters from different branches at intermediate layers of consecutive pointwise convolutions. Our experiments with the CIFAR-10 dataset show that our optimized EfficientNet has similar learning capacity to the original layout when training from scratch.

Keywords. EfficientNet, Deep Learning, Computer Vision, Classification, CNN, DCNN

1. Introduction

Plenty of image classification architectures are tested and benchmarked with ImageNet [1] dataset. On the other hand, it should be noted that not all problems in image classification have 1000 classes and millions of samples nor every research group has the required computing resources to train deep neural network models on large datasets. In this sense, we propose a highly parameter-efficient DCNN architecture that performs well at training from scratch with small datasets and small computing resources. As an example, in the scope of plant disease classification, the Cropped-PlantDoc dataset [2] has less than 10k image samples. Due to its small sample size, this dataset is prone to overfitting. We show that our highly parameter-efficient architecture performs better than the baseline when training them with small datasets.

This article is structured as follows: section 2 presents and discusses relevant work in regards to DCNNs, parameter-efficient DCNNs and datasets used in this work. Section 3 presents our proposed pointwise convolution replacement. Our results and discussion are given in sections 4 and 5, respectively. Section 6 summarizes the paper.

2. Related work

In 1980, Fukushima [3] devised a layered artificial neural network for image classification inspired by the visual cortex structure. In that network, the first layer contains neurons that detect simpler patterns with a small receptive field. Deeper layers detect more complex patterns with wider receptive fields by composing patters from previous layers. That was the first Convolutional Neural Network (CNN).

In 2012, Krizhevsky et al. [4] reported a major breakthrough in the ImageNet Large Scale Visual Recognition Challenge, using their AlexNet architecture. Since then, many other DCNN architectures have been introduced, like ZFNet [5], VGG [6], GoogLeNet [7], ResNet [8] and DenseNet [9]. Since the number of layers of proposed convolutional neural networks have increased from 5 to more than 200, those models are usually referred as Deep Learning or DCNN.

In regards to datasets, the following three are in our interest:

- The Oxford-IIIT Pet dataset [10] consists of 25 breeds (classes) of dogs and 12 breeds (classes) of cats. In total, there are 37 classes of images. Each class has around 200 images. Images have various sizes and complex backgrounds and illumination patterns.
- CIFAR-10 dataset [11] consists of 60k 32x32 images belonging to 10 different classes: airplane, automobile, bird, cat, deer, dog, frog, horse ship and truck. These images are taken from natural and uncontrolled lightning environment. They contain only one prominent instance of the object to which the class refers to. The object may be partially occluded or seen from an unusual viewpoint.
- Cropped-PlantDoc dataset [2] was devised to allow plant leaf disease classification. It was created by cropping individual leafs from a smaller dataset called PlantDoc that contained multiple leafs per image. This dataset has 13 plant species and 27 classes for different diseases on each specie. Images have complex backgrounds and the area covered by the leafs has varying sizes.

These datasets offer together an interesting broad set of classes. Cropped-PlantDoc contains plants only. The Oxford-IIIT Pet dataset contains animals only. The CIFAR-10 dataset contains animals and man made objects. Besides, they are relatively small and allow easy replication of our ideas with affordable hardware and small computing time.

To be able to reduce the number of weights in our DCNNs, we propose the use of grouped convolutions. A grouped convolution evenly separates input channels and neurons for each group. Each neuron processes only input channels entering its own group. This drastically reduces the number of weights and floating point computations. A depthwise convolution is an extreme case which each group has only one input channel and only one filter. In a depthwise convolution, the number of input channels, filters and groups are the same. In AlexNet, due to implementation constraints, convolutions were separated into two groups. In 2016, Ioannou at al. [12] experimented grouped convolutions with 2, 4, 8 and 16 groups per convolution for CIFAR-10 classification. Ioannou at al. showed that replacing 3x3 and 5x5 common convolutions by grouped convolutions can reduce the number of parameters by more than 50%. They also showed that their split architectures can keep the original classification accuracy or even improve it slightly. In their work, there was no attempt to optimize the 1x1 pointwise convolutions, i.e., convolutions that have 1x1 kernels with one trainable parameter per input channel. These kernels do not take into account neighboring positions such as, for example, 3x3 filters.

In 2017, Howard at al. [13] developed an architecture called MobileNet. The MobileNet building block is composed by a depthwise separable convolution followed by a pointwise convolution. MobileNets are parameter-efficient when compared to previous models. As an example, MobileNet-160 has nearly 45 times less parameters than AlexNet and achieves similar accuracy when classifying the ImageNet dataset. MobileNet-224 has nearly 40% less parameters than GoogLeNet and achieves higher accuracy than GoogLeNet. Howard at al. noted that as their models are more parameter-efficient, these smaller models require less data augmentation. There is an aspect in their MobileNet models that has central interest for our proposal: nearly 75% of the parameters and 95% of multiplications and additions are computed by pointwise convolutions. This makes a strong case for an optimized pointwise convolution.

Also in 2017, Ting Zhang et. al. [14] proposed mixing grouped convolutions with interleaving layers. They proposed a grouped spatial convolution followed by an interleaving layer and a grouped pointwise convolution. The most evident difference from their work to ours is about us developing a solution specifically targeting a pointwise convolution replacement.

In 2019, Mingxing Tan et al. [15] developed the EfficientNet architecture. At that time, their EfficientNet-B7 variant was 8.4 times more parameter-efficient and 6.1 times faster than the best existing architecture achieving 84.3% top-1 accuracy on ImageNet. As in the case of MobileNets, more than 80% of the parameters of EfficientNets come from standard pointwise convolutions. This aspect opens an opportunity for a huge reduction in number of parameters and floating point operations.

3. Methodology

Latest DCNN architectures have a big portion of their parameters located in pointwise convolutions. Thus, we propose replacing pointwise convolutions by parameter-efficient counterparts. Figure 1 shows a diagram of our proposed pointwise replacement. It starts with a pointwise grouped convolution K (parallel groups K_1 to K_{N_i}) followed by a channel interleaving layer which mixes channels for the next pointwise grouped convolution L (parallel groups L_1 to L_{N_i}). All channels from parallel groups K_1 to K_{N_i} are concatenated into a single output of the K layer. The same happens for the L layer. Concatenated outputs from K and L layers are summed channel by channel, which makes the L layer to behave as a residual convolution.

In standard pointwise convolutions, each filter has one trainable parameter per input channel. Therefore, the number of parameters P in layer i is calculated from the number of channels of the preceding activation map C_{i-1} and the number of filters F_i as in Eq. 1:

$$P_i = C_{i-1} \cdot F_i \tag{1}$$

For grouped convolutions, let us note the number of groups in layer i as N_i. Each group is fed a contiguous subset of C_{i-1}/N_i of the input channels. Moreover, the number of filters per group is F_i/N_i. Thus, the number of parameters per group is the number of filters per group multiplied by the number of channels per group $(F_i/N_i) \cdot (C_{i-1}/N_i)$. Therefore, multiplying the previous expression by the number of groups, we obtain the total number of parameters of a grouped convolutional layer as in Eq. 2:

$$P_i = (C_{i-1} \cdot F_i)/N_i \tag{2}$$

When grouping convolutions, we follow these constraints:

- Each group must have at least 16 input channels. This will allow a minimum degree of intragroup combinations.
- The number of groups N_i shall be the greatest common divisor of the number of input channels C_{i-1} and the number of filters F_i respecting the previous constraint $(C_{i-1}/N_i \geq 16)$.
- Given above constraints, if we cannot have more than one group, then the original pointwise convolution is not replaced by this sub-architecture.
- Only when the number of output channels per group (F_i/N_i) is bigger than 1, an interleaving layer will be added.
- Only when the number of input channels is greater or equal than the number of output channels $(C_{i-1} \geq C_i)$, a grouped pointwise convolutional layer is added after the interleaving layer and then the result of both grouped convolutional layers are summed.

As an example, in a monolithic pointwise convolution with $C_{i-1} = 1{,}024$ and $F_i = 512$, we will obtain $P_i = 524{,}288$ parameters. If we replace this pointwise convolution with our sub-architecture, according to the constraint of a minimum of 16 channels per group, N_i must be 64. In this example, the first grouped convolutional layer will have $1{,}024 \cdot 512/64 = 8{,}192$ parameters. The second grouped convolutional layer will also have 64 groups, but the number of input channels will be 512, hence the total number of parameters will be $512 \cdot 512/64 = 4{,}096$. Summing both results, the total number of parameter of the whole sub-architecture will be 12,288, which is a saving of almost 97.7% from the original parameter count.

In all of our experiments, we used an Amazon AWS instance with a single Tesla T4 GPU paired with 8 virtual cores. In regards to software, we used Keras/Tensorflow and RMSProp optimizer. Our experiment with Oxford-IIIT Pet dataset has cyclical learning rate of 25 epochs. All training experiments have data augmentation. For each dataset, we used a specific number of epochs as shown in Table 1. We used more epochs in smaller datasets to compensate the smaller number of samples. The number of epochs fits a multiple of our learning rate cycle. In this work, we did not use transfer learning as our goal is to evaluate learning capacity of parameter-efficient models. In all experiments, we kept the same dropout as per original EfficientNet (baseline) implementation.

Our source code is publicly available and can be fount at `https://github.com/joaopauloschuler/kEffNet/`.

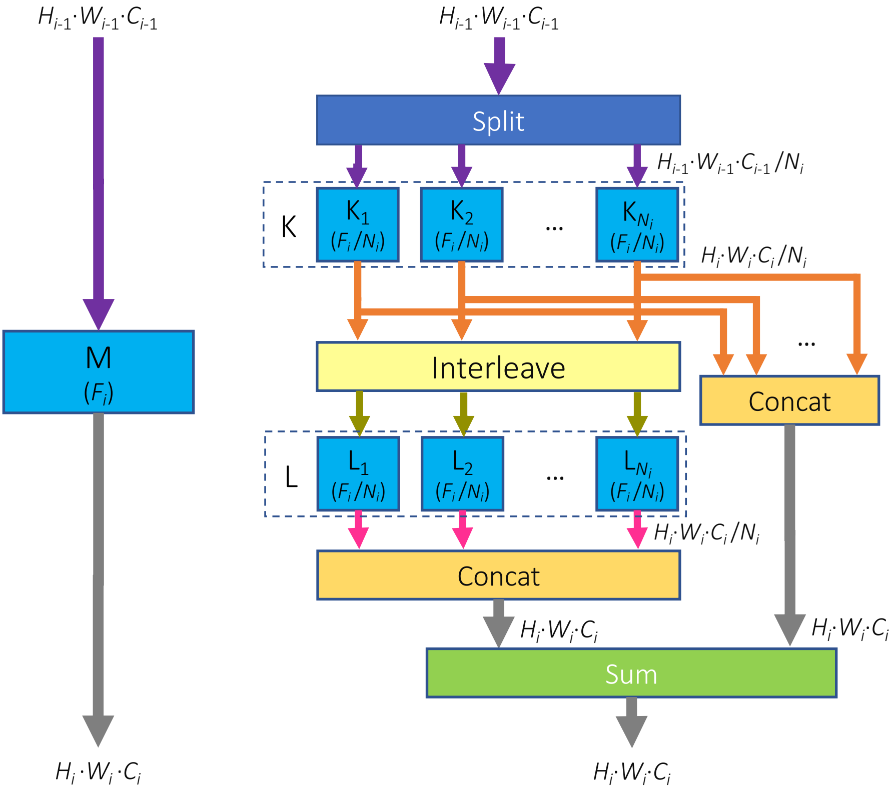

$H_{i-1} \cdot W_{i-1} \cdot C_{i-1}$
M
(F_i)
$H_i \cdot W_i \cdot C_i$
$H_{i-1} \cdot W_{i-1} \cdot C_{i-1}$
Split
$H_{i-1} \cdot W_{i-1} \cdot C_{i-1} / N_i$
K
K_1 (F_i/N_i)
K_2 (F_i/N_i)
...
K_{N_i} (F_i/N_i)
$H_i \cdot W_i \cdot C_i / N_i$
Interleave
...
Concat
L
L_1 (F_i/N_i)
L_2 (F_i/N_i)
...
L_{N_i} (F_i/N_i)
$H_i \cdot W_i \cdot C_i / N_i$
Concat
$H_i \cdot W_i \cdot C_i$
$H_i \cdot W_i \cdot C_i$
Sum
$H_i \cdot W_i \cdot C_i$

Figure 1. Graphical representation of our pointwise convolution replacement. At the left, a classic monolithic layer M with F_i pointwise filters. At the right, our substitute for M, made of two grouped pointwise convolutional layers, K and L, with N_i parallel groups in each layer. Each parallel group has F_i/N_i filters. The size of activation maps, transmitted through arrows, is the multiplication of height H, width W and number of channels C, sometimes divided by N_i. The subindex i indicates architecture level. For pointwise convolutions, H_i=H_{i-1}, W_i=W_{i-1} and C_i=F_i.

4. Results

We have analyzed two types of results: test classification accuracy and heatmaps. In the following subsections, we name our modified EfficientNet variants as ”kEffNet” followed by the corresponding complexity code ”-Bn” (”-B0”, ”-B1”, etc.).

4.1. Test Classification Accuracy

Table 1 compares test accuracies and trainable parameter count of the baseline EfficientNet-B0 and our variant kEffNet-B0 with the tested datasets.

As the scope of this work is limited to small datasets and small architectures, we only experimented with the smallest EfficientNet variant (EfficientNet-B0) and our modified variant (kEffNet-B0). Nevertheless, Table 2 provides the number of trainable parameters of the other EfficientNet variants (original and modified).

dataset	classes	epochs	model	parameters	test accuracy
Cropped-PlantDoc	29	75	EfficientNet-B0	4,044,697	49.08%
			kEffNet-B0	664,041	**64.39%**
Oxford-IIIT Pet	37	150	EfficientNet-B0	4,054,945	56.73%
			kEffNet-B0	674,289	**60.09%**
CIFAR-10	10	50	EfficientNet-B0	4,020,358	**91.66%**
			kEffNet-B0	639,702	91.64%

Table 1. Classification accuracies found with baseline and our variant.

variant	EfficientNet Parameters	kEffNet Parameters	Reduction
B0	4,020,358	639,702	84.09%
B1	6,525,994	922,770	85.86%
B2	7,715,084	1,130,344	85.35%
B3	10,711,602	1,532,902	85.69%
B4	17,566,546	1,952,306	88.89%
B5	28,361,274	2,633,470	90.71%
B6	40,758,754	4,714,030	88.43%
B7	63,812,570	4,669,218	92.68%

Table 2. Number of trainable parameters per EfficientNet and kEffNet variants, for a 10 classes dataset.

4.2. Heatmap

We produced heatmaps for a number of samples from the Oxford-IIIT Pet as shown in Figure 2. We observed that our kEffNet-B0 tends to concentrate feature activation close to the ears and to the top of the cats head, while the baseline frequently finds feature activation in the background.

In Figure 3, we find an image sample that both the baseline and our model do not focus on the cat. We speculate that the proposed model is detecting textile patterns that are commonly found in cat photos.

5. Discussion

We face two main limitations when applying grouped convolutions:

1. It prevents the DCNN from exploring all possible combinations among all features coming from the previous layer due to missing connections.
2. The output of parallel groups must be somehow joined.

To alleviate the first and partially the second limitation, we interleave the channels of the first grouped pointwise convolution before feeding the next grouped convolution. To alleviate the second limitation, we propose the use of summation operator for joining paths. Compared to concatenation, summation has the advantage of not increasing the number of output channels. Our pointwise parameter-efficient replacement does not directly explain why we got higher classification accuracy in the Cropped-PlantDoc dataset or in the Oxford-IIIT Pet dataset. Most likely, as we have less trainable parameters, our modified kEffNets are less prone to overfitting. As CIFAR-10 is comparatively a bigger dataset, we found almost the same test classification accuracy in baseline and modified

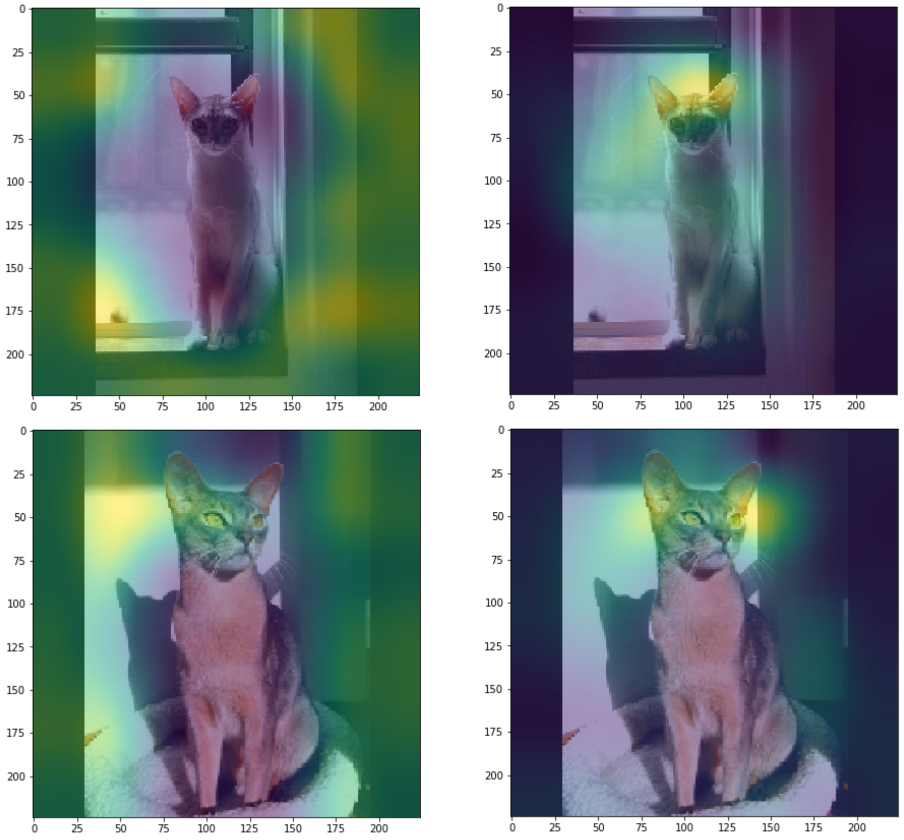


Figure 2. On the left, heatmaps produced by the EfficientNet-B0 baseline. On the right, heatmaps produced with our kEffNet-B0.

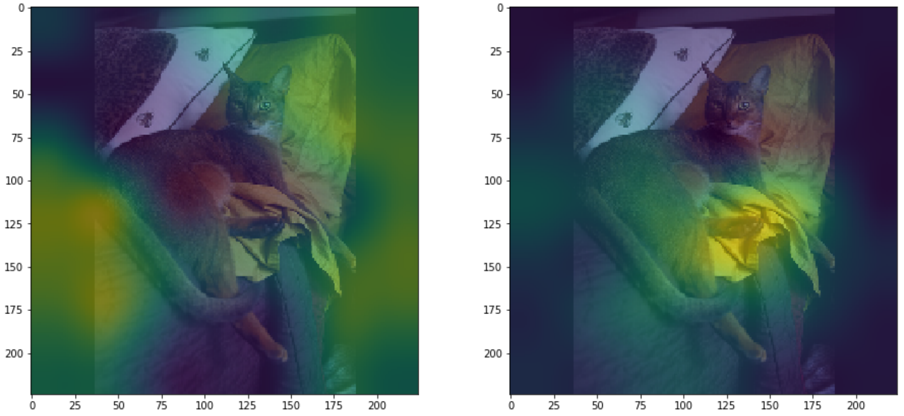


Figure 3. On the left, heatmap produced by the EfficientNet-B0 baseline. On the right, heatmap produced with our kEffNet-B0.

kEffNet. In regards to heatmaps, the baseline seems to be more frequently concentrating attention on the background. This effect may not be necessarily a product of overfitting.

6. Conclusion

In this work, we have experimented with a replacement of EfficientNet pointwise convolutions using a sub-architecture that contains up to two grouped convolutions, an interleaving layer and a summation layer at its end. By doing this replacement in the EfficientNet-B0 variant, we were able to save more than 84% of the trainable parameters while keeping the classification accuracy on the CIFAR-10 dataset when training from scratch. As we have less trainable parameters, we found significantly better classification accuracy in the Cropped-PlantDoc (+15.3%) and Oxford-IIIT Pet datasets (+3.3%) probably due to less overfitting. As a matter of future research, we may delve in fine tuning our sub-architecture details such as decreasing the dropout rate as we having less trainable parameters.

References

[1] Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV). 2015;115(3):211-52.

[2] Singh D, Jain N, Jain P, Kayal P, Kumawat S, Batra N. PlantDoc: A Dataset for Visual Plant Disease Detection. CoRR. 2019;abs/1911.10317. Available from: http://arxiv.org/abs/1911.10317.

[3] Fukushima K. Neocognitron: A Self-Organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position. Biological Cybernetics. 1980;36:193-202.

[4] Krizhevsky A, Sutskever I, Hinton GE. ImageNet Classification with Deep Convolutional Neural Networks. In: Pereira F, Burges CJC, Bottou L, Weinberger KQ, editors. Advances in Neural Information Processing Systems 25. Curran Associates, Inc.; 2012. p. 1097-105. Available from: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.

[5] Zeiler MD, Fergus R. Visualizing and Understanding Convolutional Networks. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T, editors. Computer Vision – ECCV 2014. Cham: Springer International Publishing; 2014. p. 818-33.

[6] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In: Bengio Y, LeCun Y, editors. 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings; 2015. Available from: http://arxiv.org/abs/1409.1556.

[7] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition; 2015. p. 1-9.

[8] He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition; 2015. Cite arxiv:1512.03385Comment: Tech report. Available from: http://arxiv.org/abs/1512.03385.

[9] Huang G, Liu Z, Weinberger KQ. Densely Connected Convolutional Networks. CoRR. 2016;abs/1608.06993. Available from: http://arxiv.org/abs/1608.06993.

[10] Parkhi OM, Vedaldi A, Zisserman A, Jawahar CV. Cats and Dogs. In: IEEE Conference on Computer Vision and Pattern Recognition; 2012. .

[11] Krizhevsky A. Learning multiple layers of features from tiny images; 2009.

[12] Ioannou Y, Robertson DP, Cipolla R, Criminisi A. Deep Roots: Improving CNN Efficiency with Hierarchical Filter Groups. CoRR. 2016;abs/1605.06489. Available from: http://arxiv.org/abs/1605.06489.

[13] Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. CoRR. 2017;abs/1704.04861. Available from: http://arxiv.org/abs/1704.04861.

[14] Zhang T, Qi G, Xiao B, Wang J. Interleaved Group Convolutions for Deep Neural Networks. CoRR. 2017;abs/1707.02725. Available from: http://arxiv.org/abs/1707.02725.

[15] Tan M, Le QV. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. CoRR. 2019;abs/1905.11946. Available from: http://arxiv.org/abs/1905.11946.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210159

Promising Depth Map Prediction Method from a Single Image Based on Conditional Generative Adversarial Network

Saddam ABDULWAHAB [a,1], Hatem A. RASHWAN [a], Armin MASOUMIAN [a], Najwa SHARAF [a] and Domenec PUIG [a]

[a] *DEIM, Universitat Rovira i Virgili, 43003 Tarragona, Spain*

Abstract. Pose estimation is typically performed through 3D images. In contrast, estimating the pose from a single RGB image is still a difficult task. RGB images do not only represent objects' shape, but also represent the intensity that is relative to the viewpoint, texture, and lighting condition. While the 3D pose estimation from depth images is considered a promising approach since the depth image only represents objects' shape. Thus, it is necessary to know what is the appropriate method that can be used for predicting the depth image from a 2D RGB image and then to use for getting the 3D pose estimation. In this paper, we propose a promising approach based on a deep learning model for depth estimation in order to improve the 3D pose estimation. The proposed model consists of two successive networks. The first network is an autoencoder network that maps from the RGB domain to the depth domain. The second network is a discriminator network that compares a real depth image to a generated depth image to support the first network to generate an accurate depth image. In this work, we do not use real depth images corresponding to the input color images. Our contribution is to use 3D CAD models corresponding to objects appearing in color images to render depth images from different viewpoints. These rendered images are then used as ground truth and to guide the autoencoder network to learn the mapping from the image domain to the depth domain. The proposed model outperforms state-of-the-art models on the publicly PASCAL 3D+ dataset.

Keywords. Deep learning, depth prediction, image segmentation, UNet, UNet++, image to image translation.

1. Introduction

Inferring depth and 3D pose estimation from a single RGB image is one of the most challenging problems in computer vision. In this work, we address the challenging problem of depth estimation that is useful to determine pose estimation showing in a scene using a monocular camera. Practicality, the important challenge in a 3D pose estimation directly from a single image is the ambiguity of the object shape in a monocular image. Since the appearance of an object in an image dramatically depends on its intrinsic characteristics (e.g., texture and color/albedo), and extrinsic characteristics related to the

[1] Corresponding Author: Saddam Abdulwahab. E-mail:saddam.abdulwahab@gmail.com

acquisition (e.g., camera pose and gamma correction conditions). Thus, estimating a 3D pose requires the depth image that only contains the shapes of the objects in order to estimate the correct pose of the main objects in a 3D scene.

Nowadays, with the significant progress of deep learning models, several approaches based on deep networks have been proposed to predict depth maps from a single image. In particular, [1] presents a framework for depth and surface normal estimation from monocular images. It consists of a regression stage using a deep CNN model to learn the mapping from multi-scale image patches to depth or surface normal values at the super-pixel level (the SLIC algorithm [2] that is used to segment the depth images into super-pixels). [1] used then refined the estimated super-pixel depth or surface normal to the pixel level by exploiting the potentials on the depth or surface normal map, which include a data term, a smoothness term among super-pixels and an auto-regression term characterizing the local structure of the estimated depth map. In [3], a three-layer CNN trained with a per-pixel Euclidean loss was presented to convert the given color image to a geometrically meaningful output image. Besides, they used Conditional Random Fields (CRF) as a loss layer to enforce local consistency in the output image.

This work is close in spirit to that of [4,5,6] in the sense that we also use a deep learning approach to retrieve depth maps from a single image. Our method is a promising method since it can be applied to predict depth images for indoor and outdoor scenarios. Besides, it can be used as a co-representation method to be applied to predict the pose estimation from a single RGB image. Furthermore, it is producing promising results with a high precision rate and an acceptable computational cost.

In this work, we propose to use an autoencoder network as a generator based on UNet and UNet++ models [7,8]. In particular, a cutting-edge technique for image transformation as a baseline network for predicting a depth image from a single color image. However, with the lack of annotated training data for depth images of objects, we use 3D CAD models for rendering depth images from different viewpoints. The obtained depth images are used to train the autoencoder network. The proposed model consists of two successive networks. The first network is depth estimation that learns to map the RGB image domain into the depth image domain. In order to enforce the generator to generate a depth close to the ground truth, we propose a second network is a discriminator network that helps the first network by comparing the ground truth and generated depth images. The two networks are integrated into a single pipeline to solve the problem of depth image estimation. Figure 1 shows the proposed framework for depth estimation from a single image using technique segmentation.

To the best of our knowledge, this work is the first attempt to use an autoencoder network used for the image segmentation purpose as a generator to a training model for estimating depth maps of the main object depicted in a 2D image. Consequently, the main contributions of this paper are the following:

- We propose an autoencoder segmentation network as a generator that can predict a depth image from a single 2D color image of an object.
- We propose a discriminator network to achieve a more accurate comparison of the ground truth and generated depth to enforce the autoencoder network to generate an accurate dense depth image.
- The integration of the two networks into a single pipeline to solve the problems of generating a depth image from a single color image.

This paper is organized as follows. Section 2 describes the proposed methodology to estimate a depth image using segmentation. Section 3 describes experimental results. Finally, Section 4 concludes this work and suggests future lines of research.

2. Methodology

This section explains the proposed scheme, the tools, and the resources being used in this work. We formulate the problem in subsection 2.1. The remaining subsections explain each part of the proposed model in detail.

2.1. Problem Formulation

Let $a \in A$ be a 2D color image, and the problem of generating its corresponding depth image, $b \in B$, can be dened formally as a function $f : A \rightarrow B$ maps elements from domain A to ones in its co-domain B. Figure 1 shows the graphical description of the system. It contains two main modules. The first one is a depth generator G based on an autoencoder segmentation Network, and the second one is the discriminator network D based on a CNN.

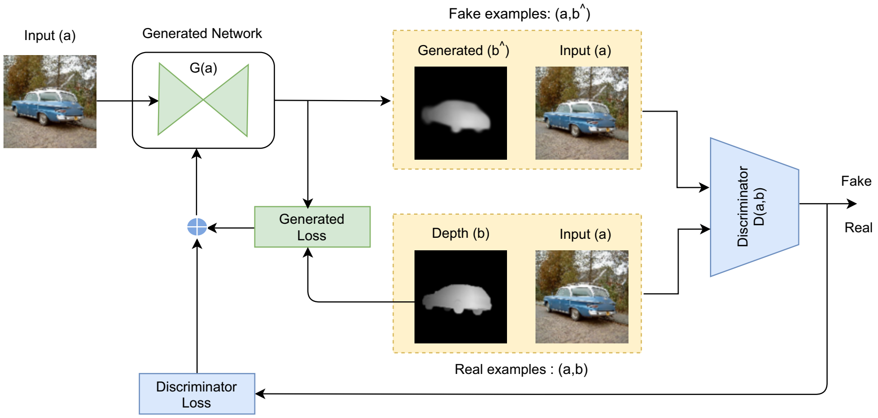


Figure 1. General overview of the proposed depth estimation model.

2.2. Generator Network

Two main variations of our autoencoder segmentation network are proposed in this paper as a generator network. Both of them are encoder-decoder neural network architecture. The first network is UNet [7], it involves convolution layers, and it does not include a fully connected layer that is demanding on a big amount of data. This network is simple, efficient, and easily used. It consists of two parts: the first one is an encoder that obtains different image feature levels continuously sampled through multiple convolution layers. Also, we tested the UNet++ [8], which consists of a series of nested dense convolutional blocks, as a encoder to choose the best between UNet and UNet++ networks

The second one is a decoder that performs multi-layer deconvolution on the top-level feature map and combines different feature levels in the down-sampling process to restore the feature map to the original input image size and completes the end-to-end depth estimation task from the input image. Besides, it uses the skip connection operation to connect each pair of down-sampling layers and the up-sampling layer that makes the spatial information directly applied to much deeper layers and a more accurate segmentation result.

The generator G learns the mapping from an input color image to the corresponding depth image. The input to the segmentation network is a 2D color image, a, and it generates a depth image, $\hat{b}$.

In order to optimize the structural similarity between the depth image and ground truth, we use two loss functions: the first one is a *MSE* loss function based on feature matching that can be defined as follows (1):

$$\mathfrak{L}_{\mathfrak{gan}}(a,b,G(a)) = \frac{1}{n}\sum_{i\in T}^{n} f(\hat{b}_{(i)} - b_{(i)})^2, \tag{1}$$

where a is input 2D color image, G is a generator network, f is the *MSE* error, $b_{(i)}$ is the real depth of pixel i, $\hat{b}_{(i)}$ is the associated predicted depth by generator network, T is the set of valid pixels (i.e., both the ground-truth and predicted depth pixels that do not have depth values equal to zero or non-black regions as shown in Figure 2 and n is the cardinality of T.

2.3. Discriminator Network

The generator network generates depth images b that belong to domain B from the domain A of color images. To model this additional constraint, we proposed a discriminator network that is composed of five convolution layers with 4×4 filters, stride 2 and padding 1. Each convolution layer is followed by batch normalization (BN) except for the first convolutional layer C_{n1} followed by an output logistic unit*LeakyReLU* [3,9]. The idea of our approach is to train the generator to generate samples very close to the real samples and the samples have to be in the depth image domain. To model this additional constraint, we train a discriminator neural network D to distinguish between a real sample consisting of (input color image and real depth image rendered from 3D CAD models) and a fake sample consisting of(input image and generated depth image from the generator G). The discriminator network D is used to determine whether the depth images estimated by the generator G are comparable to depth images or not.

In addition, it provides a second loss measure, along with the reconstruction error of the generated depth map, that is useful for training an accurate generator to generate a dense depth image and minimize the difference between the corresponding features and avoid the over-fitting and make the training more stable and converge faster. The discriminator is trained by minimizing the following binary cross-entropy (BCE) loss is defined as follows (2):

$$\ell_{Dis}(D,a,b,\hat{b}) = -\mathbb{E}_{a\hat{b}b}[log(D(a,b)) + log(1 - D(a,\hat{b})] \tag{2}$$

2.4. Total Loss

The final objective function, i.e. the training loss, at one iteration of our learning algorithm is defined as:

$$\mathfrak{L}(G,D,a,b,\hat{b}) = \ell_{gan}(G,D,a,G(a)) + \ell_{Dis}(D,a,b,\hat{b}) \tag{3}$$

This loss $\mathfrak{L}(G,D,a,b,\hat{b})$ is efficiently integrated into the back-propagation for the generator network through ADAM optimization.

3. Experiments and Results

This section describes the experiments performed to evaluate the proposed model on the publicly PASCAL 3D+ dataset using various evaluation measures.

3.1. Dataset

In this work, a comprehensive set of experiments have been conducted to validate the performance of the proposed model on the public PASCAL3D+ dataset [10], which contains 12 object categories. Every object category contains ten or more 3D models and more than $1,000$ color images related to every category. We used the 3D models to render corresponding depth images for the RGB images to train the proposed model. We render a depth image from a 3D model corresponding to each color image according to the viewpoints specified in the dataset. We randomly split the images in each category into 70% for the training set and 30% for the testing set. To increase the number of training samples, we apply data augmentation (DA) techniques Shown in Figure2 that show the transformations applied to every input image and the corresponding depth images. Thus, each category has more than $10,000$ images for training the model. After applying data augmentation to the real color images and corresponding depth ones, and using them as inputs to the model for the training process, we found that the efficiency of the network significantly improved due to exposing the model with more difficult samples and samples under different conditions. For all the tested 3D models, we rendered depth images using the MATLAB 3D Model Renderer [2] based on the viewpoints (i.e, azimuth and elevation angles), as well as the distance between the camera and the 3D model obtained from the annotation of the PASCAL 3D+ dataset.

3.2. Parameter settings

We used the Adam optimizer [11] with $\beta_1 = 0.5$, $\beta_2 = 0.999$ and an initial learning rate of 0.0001. A batch size of 4 with 1000 epochs yielded the best combination. The input images is reshaped to 128×128 pixels and normalizes through divided by 255. For all these experiments, we used a 64-bit I7-6700, 3.40GHz CPU with 16GB of memory, as well as one NVIDIA GTX 1080 GPU on Ubuntu 16.04. We used the Pytorch [12] deep

[2] https://www.openu.ac.il/home/hassner/projects/poses/

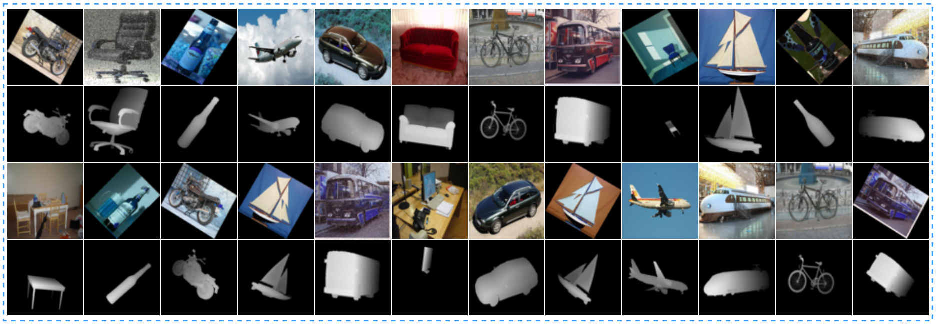


Figure 2. Transformations (flipping, blurring, noise, and rotation) are applied to every real image and its corresponding rendered depth image in all transformations, except blurring and noise, we apply them for the real image only.

learning framework. The computational time of the proposed method for the training process takes around 1.2 minutes for each epoch with a batch size of 4. In turn, the online estimation of depth maps has a performance of around 7 images per second.

3.3. Evaluation Measures

For depth image prediction, we used four different measures to assess the final performance. The first measure is the root mean square error (RMSE), which provides a quantitative measure of per-pixel error, computed as (4):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i\in T}(\hat{b}_{(i)} - b_{(i)})^2}, \tag{4}$$

where $b_{(i)}$ is the real depth of pixel i, $\hat{b}_{(i)}$ is the associated predicted depth, T is the set of valid pixels (i.e., both the ground-truth and predicted depth pixels that do not have depth values equal to zero or non-black regions as shown in Figure 2 and n is the cardinality of T.

The second measure assesses the accuracy of the proposed model to estimate errors under a given threshold, serving as an indication of how often our estimate is correct. The threshold accuracy measure from [3] is essentially the expectation that the depth value error of a given pixel in T is lower than a threshold thr^Z:

$$\delta_Z = \mathbb{E}_T[F(\max(\frac{b_{(i)}}{\hat{b}_{(i)}}, \frac{\hat{b}_{(i)}}{b_{(i)}}) < thr^Z)] , \tag{5}$$

where $F(\cdot)$ represents an indicator function that yields 0 or 1. As in [3], we set $thr = 1.25$, and $Z \in \{1,2,3\}$.

The third measure is the Intersection Over Union (IOU) value, also referred to as the Jaccard index that can be computed as (6):

$$IoU = \frac{TP}{TP+FP+FN}, \tag{6}$$

where TP indicates the number of pixels whose estimated depth coincides with the real depth, FP indicates the opposite, and FN indicates the number of pixels where the real depth has no predicted depth associated.

The fourth measure is the Dice score, which computes the ratio between the amount of intersection and the total number of pixels in both the predicted $\hat{b}$ and the real depth b that can be defined as (7):

$$Dice = \frac{2|\hat{b} \cap b|}{|\hat{b}| + |b|} = \frac{2TP}{2TP + FP + FN}. \tag{7}$$

Table 1. Results for depth image estimation from 2D color images on the PASCAL3D+ dataset under different measures with (a) GAN proposed in [13], (b) GAN with a reconstruction loss proposed in [14], (c) Adversarial Learning proposed in [6] and (d) the proposed model. Lower is better for the RMSE metric, and higher is better for the other measures. The best results are highlighted in bold.

			aero	bike	boat	bottle	bus	car	chair	table	mbike	sofa	train	tv	mean
GAN Model		IoU	0.46	0.26	0.50	0.80	0.62	0.71	0.42	0.44	0.48	0.61	0.56	0.78	0.55
		Dice	0.62	0.40	0.66	0.87	0.76	0.82	0.57	0.60	0.61	0.73	0.71	0.87	0.69
		RMSE (linear)	0.21	0.26	0.20	0.14	0.16	0.18	0.23	0.23	0.24	**0.15**	0.19	0.15	0.20
		threshold $\delta < 1.25$	0.77	0.53	0.69	0.66	0.47	0.63	0.71	0.75	0.65	0.59	0.56	0.53	0.63
		threshold $\delta < 1.25^2$	0.83	0.60	0.78	0.83	0.63	0.74	0.81	0.81	0.71	0.78	0.68	0.69	0.74
		threshold $\delta < 1.25^3$	0.86	0.65	0.84	0.89	0.73	0.83	0.85	0.83	0.77	0.87	0.75	0.81	0.81
GAN with a reconstruction		IoU	0.49	0.32	0.51	0.79	0.67	0.73	0.47	0.42	0.51	0.63	0.61	**0.80**	0.58
		Dice	0.64	0.41	0.66	0.88	0.79	0.84	0.62	0.58	0.67	0.76	0.75	**0.89**	0.71
		RMSE (linear)	0.20	0.24	0.20	0.17	**0.15**	0.18	0.23	0.23	0.22	0.18	0.18	0.16	0.20
		threshold $\delta < 1.25$	0.83	0.65	0.68	0.76	0.61	0.67	0.72	0.77	0.72	0.66	0.62	0.54	0.69
		threshold $\delta < 1.25^2$	0.87	0.68	0.76	0.85	0.75	0.79	0.80	0.84	0.80	0.80	0.69	0.71	0.78
		threshold $\delta < 1.25^3$	0.89	0.70	0.82	0.89	0.80	0.85	0.83	0.85	0.84	0.88	0.79	0.82	0.83
Adversarial Learning		IoU	0.52	**0.43**	0.56	**0.82**	0.70	0.75	0.52	0.49	0.53	0.62	0.54	**0.78**	0.61
		Dice	0.66	**0.55**	0.71	**0.90**	0.81	0.86	0.67	0.65	0.68	0.75	0.69	**0.87**	0.73
		RMSE (linear)	**0.18**	**0.23**	0.20	0.14	**0.15**	0.16	0.21	0.22	0.23	0.16	0.20	**0.14**	0.19
		threshold $\delta < 1.25$	0.80	0.70	0.65	0.76	0.58	0.69	0.74	0.78	0.71	0.61	0.58	0.52	0.68
		threshold $\delta < 1.25^2$	0.85	0.74	0.75	0.86	0.72	0.82	0.82	0.84	0.79	0.79	0.68	0.70	0.78
		threshold $\delta < 1.25^3$	0.87	0.76	0.82	0.91	0.79	0.87	0.86	**0.87**	0.84	0.86	0.76	0.81	0.84
Our Model	UNet	IoU	**0.53**	0.41	**0.61**	0.77	0.75	0.79	**0.62**	**0.53**	0.55	**0.70**	0.65	**0.78**	**0.64**
		Dice	0.69	0.51	**0.75**	0.86	**0.83**	**0.88**	**0.75**	**0.68**	0.69	**0.82**	0.77	**0.87**	**0.758**
		RMSE (linear)	**0.18**	0.25	**0.17**	**0.11**	**0.15**	**0.14**	**0.20**	**0.21**	0.22	0.20	0.17	0.17	**0.18**
		threshold $\delta < 1.25$	**0.87**	0.70	0.67	**0.82**	0.65	**0.78**	**0.78**	**0.82**	0.81	**0.84**	0.72	**0.69**	0.762
		threshold $\delta < 1.25^2$	**0.89**	0.74	0.76	**0.87**	0.78	**0.86**	**0.87**	**0.85**	**0.86**	**0.89**	0.76	**0.76**	0.824
		threshold $\delta < 1.25^3$	**0.90**	**0.78**	**0.85**	**0.93**	0.85	0.88	**0.89**	**0.87**	0.88	**0.91**	0.81	**0.86**	0.867
	UNet++	IoU	**0.53**	0.42	**0.61**	0.77	**0.76**	**0.81**	0.61	0.52	**0.56**	0.69	**0.67**	0.76	**0.64**
		Dice	**0.70**	0.51	**0.75**	0.86	**0.83**	**0.88**	**0.75**	0.67	**0.70**	0.81	**0.78**	0.85	0.757
		RMSE (linear)	**0.18**	0.24	**0.17**	0.12	**0.15**	**0.14**	**0.20**	**0.21**	**0.21**	0.21	**0.16**	0.17	**0.18**
		threshold $\delta < 1.25$	**0.87**	**0.71**	**0.69**	**0.82**	**0.67**	**0.78**	**0.78**	0.81	**0.82**	0.83	**0.73**	0.67	**0.765**
		threshold $\delta < 1.25^2$	**0.89**	**0.75**	**0.78**	**0.87**	**0.79**	**0.86**	**0.87**	0.84	**0.86**	**0.89**	**0.78**	**0.76**	**0.828**
		threshold $\delta < 1.25^3$	**0.90**	**0.78**	**0.85**	**0.93**	**0.86**	**0.89**	**0.89**	0.86	**0.89**	**0.91**	**0.83**	**0.86**	**0.87**

3.4. Results and Discussion

We have compared the proposed model with three alternative methods using the PASCAL3D+ dataset: [13,14,6]. In Table 1, we show the four evaluation measures for the predicted depth images corresponding to the 12 categories of the PASCAL3D+ dataset. We evaluate the results with three different versions of GAN and our proposed model. The first version is the GAN model proposed in [13]. The second version is the GAN model with a reconstruction loss based on the L1-norm proposed in [14]. The third version is the adversarial learning model proposed in [6]. Our model achieved the best mean results for the 12 categories with the four measures used in the evaluation. It achieved an average IOU score of 64% and a Dice score of 75.8%. In turn, the RMSE error with the proposed model is 0.18. With $\delta_Z = 1.25$, the accuracy rate is 76.5%, while with $\delta_Z = 1.25^3$, the accuracy rate is increased by 3%. That shows the effect of discriminator and feature matching on improving the performance of the estimation of depth images. However, the other three tested methods provided results better than our model, for bike, and bottle.

For a qualitative assessment, Figure 3 shows how the proposed model can generate depth images that are very close to the ground truth. The figure shows the output of the proposed model for different categories of PASCAL 3D+.

In addition, the performance of the proposed model for some of the categories of PASCAL 3D+ is shown in Figure 4. We show the depth image generated against the real depth images rendered from the corresponding 3D models. Besides, we show composite images from the color and the generated depth image in (row 1 and row 2). These examples show that the proposed model can predict a proper depth image that has the pose of the object in color images.

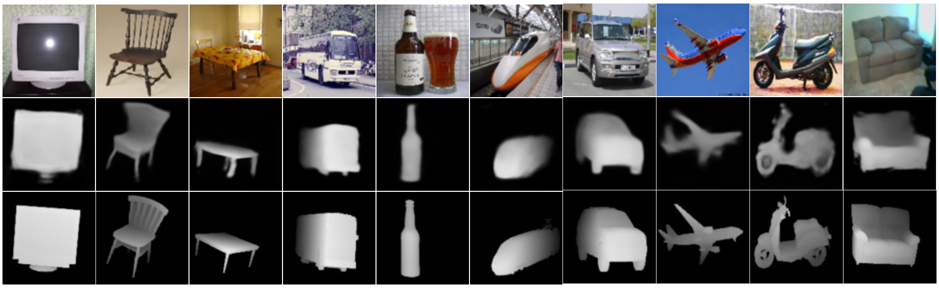


Figure 3. Intensity images (row 1), resulting depth images (row 2), Ground-truth depth images (row 3).

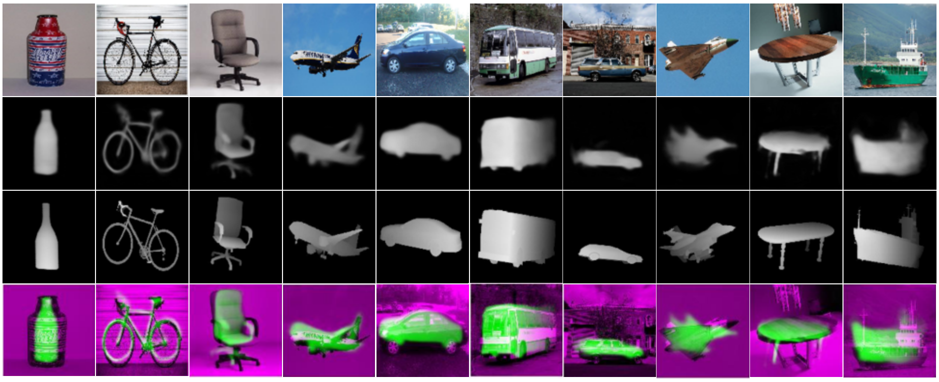


Figure 4. We show some correct and erroneous predictions given by our final method compared to the ground-truth. As it is shown, in the first six columns, we show the correct prediction, and in the last four-columns, we show the error prediction. Intensity images (row 1), resulting depth images (row 2), Ground-truth depth images (row 3), and composite images from intensity and resulting depth images (row 4).

4. Conclusion

We have introduced a novel cross-domain deep model for estimating a depth image of the main object depicted in a 2D color image. We have designed a deep model based on two deep networks. The first network is an autoencoder segmentation network, called a generator. The generator network maps the color image to a depth image. The second network is a discriminator network to achieve more comparison and allows the system to generate a dense depth image. During training, the proposed model in the first network is fed with a single 2D image for the object and the corresponding depth image rendered from a 3D model of the same object. Both the input color image and the depth image generated by the generator network are fed into the discriminator to make a more accurate comparison to the ground truth images to help in generating a more precise depth image. The model performance has been evaluated on the PASCAL3D+ dataset, yielding promising results with a high precision rate and a low computational cost comparing to the state of the art. Future work aims at applying the generated depth maps to predict the pose estimation of objects showing in a scene.

5. Acknowledgements

Financial support was given by the pre-doctoral grant (FI 2020) given by the Catalan government.

References

[1] B. Li, C. Shen, Y. Dai, A. Van Den Hengel, and M. He, “Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1119–1127, 2015.

[2] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, “Slic superpixels compared to state-of-the-art superpixel methods,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.

[3] F. Liu, C. Shen, and G. Lin, “Deep convolutional neural fields for depth estimation from a single image,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5162–5170, 2015.

[4] D. Eigen, C. Puhrsch, and R. Fergus, “Depth map prediction from a single image using a multi-scale deep network,” in *Advances in neural information processing systems*, pp. 2366–2374, 2014.

[5] D. PUIG, “Mgnet: Depth map prediction from a single photograph using a multi-generative network,” in *Artificial Intelligence Research and Development: Proceedings of the 22nd International Conference of the Catalan Association for Artificial Intelligence*, vol. 319, p. 356, IOS Press, 2019.

[6] S. Abdulwahab, H. A. Rashwan, M. A. Garcia, M. Jabreel, S. Chambon, and D. Puig, “Adversarial learning for depth and viewpoint estimation from a single image,” *IEEE Transactions on Circuits and Systems for Video Technology*, 2020.

[7] 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, Springer, 2015.

[8] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, “Unet++: A nested u-net architecture for medical image segmentation,” in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pp. 3–11, Springer, 2018.

[9] A. L. Maas, A. Y. Hannun, and A. Y. Ng, “Rectifier nonlinearities improve neural network acoustic models,” in *Proc. icml*, vol. 30, p. 3, 2013.

[10] Y. Xiang, R. Mottaghi, and S. Savarese, “Beyond pascal: A benchmark for 3d object detection in the wild,” in *Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on*, pp. 75–82, IEEE, 2014.

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

[12] A. Paszke, S. Gross, S. Chintala, and G. Chanan, “Pytorch: Tensors and dynamic neural networks in python with strong gpu acceleration,” 2017.

[13] 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*, pp. 2672–2680, 2014.

[14] T. Kim, M. Cha, H. Kim, J. Lee, and J. Kim, “Learning to discover cross-domain relations with generative adversarial networks,” *arXiv preprint arXiv:1703.05192*, 2017.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210160

Efficient Fundus Image Gradeability Approach Based on Deep Reconstruction-Classification Network

Saif KHALID [a,b,1], Saddam ABDULWAHAB [a] Hatem A. RASHWAN [a]
Julián CRISTIANO [a] Mohamed ABDEL-NASSER [a,c] and Domenec PUIG [a]
[a] *DEIM, Universitat Rovira i Virgili, Tarragona, Spain*
[b] *University of Al-Qadisiyah, Iraq*
[c] *Department of Electrical Engineering, Aswan University, 81528 Aswan, Egypt*

Abstract.
Quality of retinal image is vital for screening of ailments pertaining to eye such as glaucoma, diabetic retinopathy (DR) and age related macular degeneration. Therefore, assessing quality of retinal image prior to any kind of diagnosis has assumed significance in Computer Aided Desgin (CAD) applications. The rationale behind this is that reliability of retinal image is to be guaranteed to have dependable diagnosis. In this paper, we propose a novel retinal fundus image quality assessment (RIQA) method based on autoencoder network to assess retinal images if the image is acceptable for screening or not. The autoencoder network architecture is well suited to precisely to properly represent the key features of the image quality, especially when the network can correctly reconstruct the input image. The proposed model consists of encoder and decoder successive networks. The encoder will be used for representing the features of the input image. In turn , the decoder will be used for reconstruct the input image. The features get from encoder network will then be fed to a classifier in order to classify the quality of retinal image to two classes: gradable or ungradable. The experimental results revealed more useful assessment and the proposed deep model provides a superior performance for RIQA. Thus, our model can serve real-world Clinical Decision Support Systems in the healthcare domain.

Keywords. Retinal Image, Quality Assessment, Autoencoder Network, Ocular Diseases, Deep Learning

1. Introduction

Retinal diseases are on the rise due to the increase in the diabetic population and increased life expectancy. A good example of the effect of age on retinal health is age-related macular degeneration (AMD).The number of people having AMD disease is estimated to be 196 million in 2020, and is expected in next twenty years will increase to 288 million. In addition, Glaucoma is a progressive optic neuropathy that causes blindness in industrialized countries, and ocular hypertension is the main risk factor for glaucoma [1].

[1] Corresponding Author: Saif khalid. E-mail: saif.khalid@qu.edu.iq

Poor quality retinal fundus images can lead to an incorrect automatic diagnosis of eye diseases and classification of its severity. Therefore, an expert must first visually classify the images as gradable or non-gradable. Visual analysis of large databases of retinal images is a time-consuming task that can be distributed among a large number of experts. However, this may lead to intervariability between the results in the quality assessment or diagnosis of different experts. Therefore, automatic analysis of retinal images is a possible solution to the lack of human experts. Moreover, to better understand the cause and progression of diseases, it may be necessary to analyze many images over a long period of time. However, the accuracy of these tasks required high quality retinal images. Automated quality analysis of retinal images can reduce the need for human intervention and create better conditions for further studies, increasing the functionality of retinal disease diagnosis based on computer aided design (CAD) systems.

Therefore, it is essential to have a measure to assess the quality of the retinal image prior to processing it for diagnosis [2]. Different techniques are found in the literature to assess the quality of retinal images. There are different metrics also used for measuring the quality of retinal images. Most of previous works depend on hand-crafted computer vision techniques, such as [3,2,4]. Nowadays with the progress of deep learning techniques, recent works use different deep learning networks architectures to develop a accurate retinal images quality assessment, such as [5,6,7,8].

All of the aforementioned eye diseases diagnosis systems scientifically depend on the quality of retinal images. Thus, there is a need for an improved approach for reliable and trustworthy retinal quality assessment for improving early detection of eye diseases. Towards this end, in this work, we propose a deep learning framework, which consists of two successive networks: an auto-encoder network depend on reconstruction image and a CNN-based classifier. The encoder will used for extract the key features related the quality of retinal images. These features get from encoder network are then fed to the classifier in order to classify the quality of the input retinal images. Our contributions in this paper are as follows:

- We propose an autoencoder network to correctly learn representative deep features of the retinal fundus images via the encoder network. The decoder part is used to reconstruct the input fundus image.
- We propose a CNN classifier fed by the features learned by the encoder network to classify the input fundus images as gradable or ungradable.
- We propose the use of the mean-square-error metric (MSE) as a loss function to train the autoencoder network. The MSE loss function calculates the sum of squared distance between input image and reconstructed image by the decoder. Also, we use a binary cross-entropy loss function to train the CNN classifier.
- We propose to integrate the losses of the two autoencoder network and the CNN classifier into a single learning framework to solve the fundus image gradeability problem.

The remainder of the paper is structured as follows. Section 2 reviews recent related literature to retinal image gradeability assessment. Section 3 presents the proposed method. Section 4 presents experimental results. Section 5 concludes the paper and gives directions for future scope of the research.

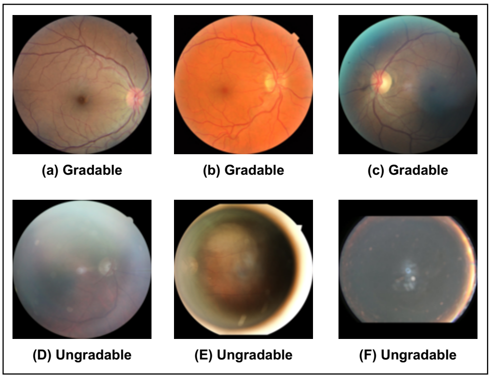


Figure 1. Example of gradable (top) and ungradable (bottom) retinal fundus images.

2. Related Work

In this section, we present a quick review of previous works related to retinal image assessment by using classical computer vision techniques and deep learning techniques. Many methods are based on binary quality labels (i.e., Accept and Reject), and others are based on a three-level quality grading system (i.e., Good, Usable and Reject). To our knowledge, no method has been proposed so far to classify the quality of retinal images through an autoencoder network like our model.

Just to name a few, [2] proposed a system for automatically detecting quality of retinal images based on the RGB color-space. Their system uses vessel density, textural features, global histogram features besides a metric known as non-reference perceptual sharpness. They also concentrated on three Regions of Interest (ROI) such as lower retinal hemispheres, upper retinal hemispheres and optic disc region. [3] also proposed a novel method for retinal image registration based on a concept named Salient Feature Region (SFR). The authors defined a measure called region saliency measure that is used to obtain SFR using gradient field entropy and local adaptive variance. They also used another method that combines gradient field distribution and computation of SFRs. However, it needs further enhancement for dealing with multimodal images. Besides, [9] introduced to Distortion Identification-based Image Verity and INtegrity Evaluation (DIIVINE) to understand quality of distorted images. That identifies distortions and then assess quality based on the estimated distortions. [10] also proposed a Blind Image Quality Assessment (BIQA) method based on gradient magnitude and Laplacian of Gaussian response. [11] proposed a method named no-reference quality assessment metric for assessing quality of video encoding. The metric has two models, coefficient distribution model resulting in local error estimation and perceptual model resulting a quality score for a test image. One of the drawbacks of this approach is that needs further enhancement to avoid transmission errors.

Based on machine and deep learning, many methods have been proposed. For instance, [12] proposed a quality assessment method based on S3D INtegrated Quality (SINQ) Predictor that based on both univariate and bivariate statistical features obtained from image. In turn, [13] proposed a method for BIQA based on learning quality lookups and receptive fields. It used both local and global receptive fields. Their method needs improvement to consider different distortions in the training images. [14] developed a framework for assessing 3D quality assessment of images containing filters to improve the image quality and algorithms to assess this quality. It is based on a technique called no-reference quality assessment. They used Artificial Neural Network (ANN) to achieve the assessment task. [5] also proposed a framework for synthesis of retinal colour images using an adversarial setting with generator and discriminator. Their framework needs further enhancement to overcome exhibiting abnormal interruptions. Recently, [15] analyzed the influences of different color-spaces on the retinal images assessment, and proposed a deep network, called Multiple Color-space Fusion Network (MCF-Net). MCF-Net integrated the different color-space representations (i.e., RGB, HSV and LAB) at a feature-level and prediction-level to predict image quality grades.

3. Proposed Model

Fig. 1 represents the training and testing phase of the proposed model. In the training model, we have uses an autoencoder network that consists of two serial networks: encoder and decoder. The encoder will be used for extracting the high-level features of the input images. The extracted features will be then fed to the decoder network to reconstruct again the same input image. Afterwards, another network, a classifier, will be fed by the features gotten from the encoder network to classify the quality of a retinal image into two classes: gradable and ungradable. The size of the input image is reshaped to 224224. In the testing model, we used only the trained encoder and the classifier network in order to classify the quality of a retinal image.

3.1. Network Architecture

Our model is based on the UNet [16] network. It is an encoder-decoder deep network architecture. UNet network is a full convolution network that does not include a fully connected layer and is not demanding on the amount of dataset. This network is simple, efficient, and easily used and adapted. It consists of two sub-netowrks, the first sub-network is an encoder that obtains different image feature levels continuously sampled through five convolution layers. The second one is a decoder that performed five deconvolution layers on the top-level feature map and combined different feature levels in the down-sampling process to restore the feature map to the original input image size and complete the end-to-end segmentation task of the image. Besides, it uses the skip connection operation to connect each pair of down-sampling layers and the up-sampling layer, which makes the spatial information directly applied to much deeper layers and a more accurate segmentation result.

The main task of using the UNet network is for semantic segmentation. However,in this paper, we used UNet for reconstructing the input image. We believe that if the autoencoder network successed in reconstruct the same input image, that means the network successed to learn the key features of the input image including visual quality features. Thus, the input to our UNet network in a RGB image and the target is the same RGB image.

For the quality classification network, we used very simple network. This classifier consists of one fully connected layer followed with Sigmoid as an activation function. This layer classify the features extracted by the encoder network to two quality labels that are gradable or ungradable.

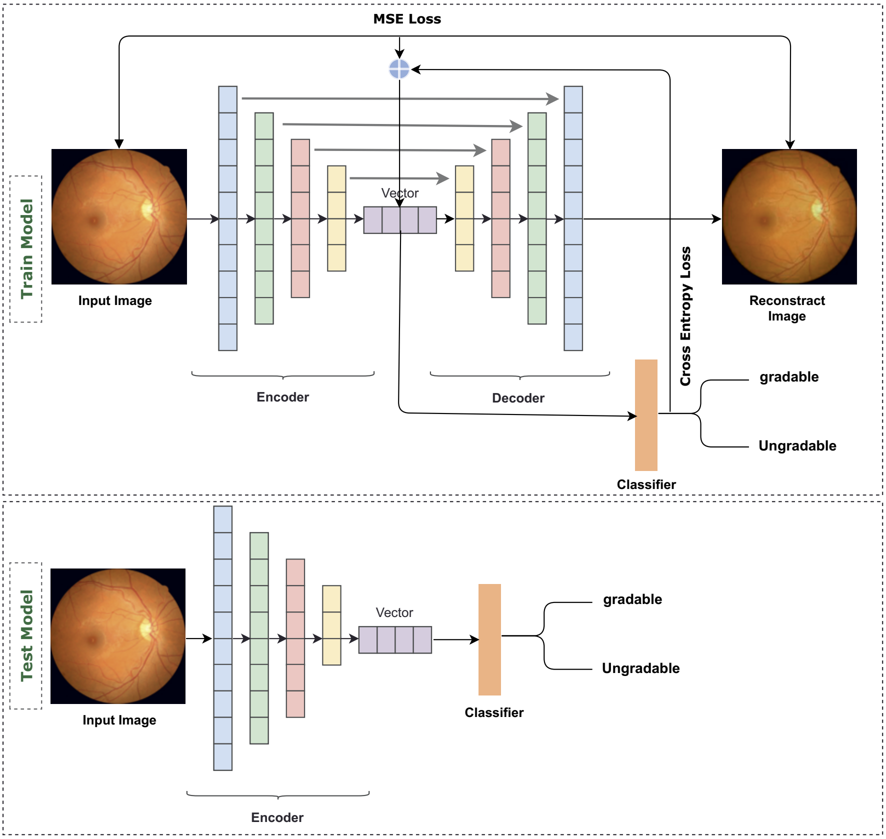


Figure 2. General overview of the proposed model in train and test stage.

3.2. Training

To assess the performance for optimizing the training of the network, we tried to use different loss function to compare the input image to the reconstructed image. This loss function is named L_{rec}. The first tested loss function is a Mean squared error (MSE) depends on the features extracted from both the input real image A and the reconstructed image $\hat{A}$ from the autoencoder network. MSE is the most commonly used a loss function for regression tasks. The loss is the mean overseen data of the squared differences between true and predicted values, MSE is defined as follows:

$$L_{rec} = \frac{1}{n}\sum_{i=1}^{n}(A_i - \hat{A}_i)^2, \quad (1)$$

where $A_{(i)}$ is the input image of pixel i, $\hat{A}_{(i)}$ is the reconstructed image and the n is the numbers of pixels in an image.

The second loss function is Mean absolute error (MAE). MAE is the mean overseen data of the absolute differences between true and predicted values, which depends on the features extracted from both the input real image A and the reconstructed image $\hat{A}$ from auto-encoder network, and it can be defined as:

$$L_{rec} = \frac{1}{n}\sum_{i=1}^{n} |A_i - \hat{A}_i|. \tag{2}$$

The third loss function structural similarity index measure (SSIM) is a method for predicting the perceived quality of digital images. SSIM is used for measuring the similarity between two images. The SSIM index is a full reference metric for measuring the quality of reconstructed images compared to input images. SSIM can be defined as:

$$SSIM(\hat{A},A) = \frac{(2\mu_{\hat{A}}\mu_A + c_1)(2\sigma_{\hat{A}A} + c_2)}{(\mu_{\hat{A}}^2 + \mu_A^2 + c_1)(\sigma_{\hat{A}}^2 + \sigma_A^2 + c_2)}, \tag{3}$$

where $\mu_{\hat{A}}$ is the mean of $\hat{A}$, $\sigma_{\mu_{\hat{A}}}$ is the standard deviations of $\hat{A}$, μ_A is the mean of A, σ_{μ_A} is the standard deviations of A, $\sigma_{\hat{A}A}$ is the covariance of $\hat{A}$ and b, $c1 = 0.01^2$, $c2 = 0.03^2$, respectively.

The fourth loss function multi-scale structural similarity index measure (MS-SSIM). The MS-SSIM approach is based on modeling of image luminance, contrast and structure at multiple scales and Multi-scale method is a convenient way to incorporate image details at different resolutions. MS-SSIM can be defined as:

$$MS-SSIM(\hat{A},A) = [L_M(\hat{A},A)]^{\alpha m} \cdot \prod_{j=1}^{m} [C_j(\hat{A},A)]^{\beta_j} \cdot [S_j(\hat{A},A)]^{y_j}, \tag{4}$$

where m represents the image quality assessment scale, which is obtained after M_i iterations.At the $j-th$ scale, the contrast comparison C_j ($\hat{A}$,A) and the structure comparison S_j($\hat{A}$,A) and The luminance comparison L_M($\hat{A}$,A), and the exponents αm,βj, and y_j are used to adjust the relative importance of different components.

For the quality labelling task, we used the standard loss function L_c, cross-entropy (CE), which depends on the predicted class from the classifier $\hat{y}$ and corresponding target value y. CE is defined as follows:

$$L_c = -\sum_{i=1}^{n} y_i \cdot log(\hat{y}_i), \tag{5}$$

where $\hat{y}_i$ is the i-th scalar value in the model output, y_i is the corresponding target value, and output size is the number of scalar values in the model output. This loss is a very good measure of how distinguishable two discrete probability distributions are from each other.

In this context, y_i is the probability that event i occurs and the sum of all y_i is 1, meaning that exactly one event may occur. The minus sign ensures that the loss gets smaller when the distributions get closer to each other.

The final objective function, during the training for the model, is defined as follows:

$$Loss = (L_{rec} + L_c)/2. \tag{6}$$

4. Experiments and Results

The experiments performed to evaluate the proposed model are described in this section, datasets and the evaluation measures used in the experiments.

4.1. Datasets

There are several publicly available RIQA datasets with manual quality annotations, such as Eye-Quality (EyeQ) [15], HRF [17], DRIMDB [18] and DR2 [19]. Among them, the EyePACS- Quality dataset is a publicly available database with about 31,032 images labelled with gradable and ungradable without Data augmentation. we selected 29,033 for the training set and 1999 for the testing set. The second (EyeQ) dataset with 28,792 retinal images with manual quality labels (good, reject). Among them, the dataset is divided into 12,543 for training and 16,249 for testing.

4.2. Data augmentation

CNN networks require large data sets in order to avoid overfitting. A class balanced dataset is also desirable as well. One of the proven approaches that can yield good results is to get more data from small datasets. In this study, we applied data augmentation techniques to the training images as proposed in [20,15] to increase the number of training samples under different conditions. Figure 3 shows the transformations applied to every input image. The applied data augmentation is done in two steps. The first steps a copy of the training examples of the small classes is done until they have the same number of images as the biggest class. This generates an equilibrated training set. In the second step, to every image, different transformations are applied in order to diversify the training, such as random uniform rotation and random flipping. Training data has two class first class (1) gradable contain 21,812 image and(0) ungradable class contain 7,218,After augmentations class 0 will be 28,872 and class 1 will be 29,083 , total for training for each class (i.e., gradable and ungradable), we have 57,957 images

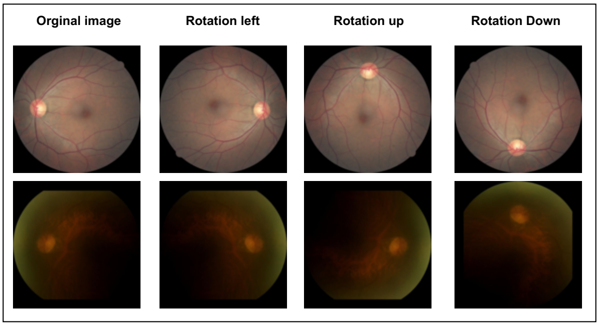


Figure 3. Transformations (flipping and rotation) applied to every real image in all transformations.

4.3. Parameter settings

We trained our networks. We used the Adam optimizer [21] with $\gamma = 0.1$, and an initial learning rate of 0.001. A batch size of 2 and 50 epochs yielded the best combination. All experiments were run on a 64-bit Core I7-6700, 3.40GHz CPU with 8GB of memory,as well as one NVIDIA GTX 1080 GPU on Ubuntu 16.04 and the PyTorch [22] deep learning framework. The computational time of the proposed method for the training process takes around 1 hour and 13 minutes and 37 second for each epoch with a batch size of 2 with train stage. In turn, the our model has a performance of around and 0.076 second for each image in train stage and 0.026 second for each image in test stage.

4.4. Evaluation Metrics

In this work, to assess the proposed model, we use five different metrics to measure the resulting performance that are Accuracy, Sensitivity (Recall), Specificity, Precision and F1 Score. In medical diagnosis, sensitivity measures the model ability to correctly identify high quality images, whereas specificity measures the model ability to correctly identify low quality images.

4.5. Results and Discussion

To evaluate the performance of the proposed model, we compared its results against a recent fundus image gradeability method called Multiple Color-space Fusion Network(MCF-Net) [15]. Both models were trained with the two datasets (i.e., subsets from Eyepacs and Eye-Quality). We tested different variations of loss functions with the proposed model: MSE, MAE, $SSIM$ and $MS-SSIM$ (Section 3) to find the best loss function that can help converge the network. Based on the two datasets, Table 1 and 2 show the results (i.e, Accuracy, Sensitivity, Specificity, Precision and $F1$ score) of MCF-Net and four variations of the proposed systems with the four loss functions. As shown in Table 1, the proposed model with its four variations outperformed the performance of MCF-Net in terms of the five evaluation matrices. Among them, our model with MSE as a loss function yielded the best results with $F1$ score, sensitivity and specificity of 0.88, 0.83 and 0.91, respectively. For instance, our model with MSE yielded an improvement of 8% with $F1$ score compared to the MCF-Net. In turn as shown in Table 2 and with the second dataset EyeQ, the proposed model and its variations also outperformed the results with MCF-Net. Our model with MSE achieved significant improvements of 16%, 10% and 38% with $F1$ score, precision and specificity, respectively. Besides, a small improvement of around 1% with sensitivity.

Table 1. Comparison between the proposed model and MCF-Net [15] on the Eypces dataset [20]

	Accuracy	Sensitivity	Specificity	Precision	F1-Score
MCF-Net Model	0.81	0.64	**0.95**	0.84	0.80
Our Model - SSIM Loss	0.815	**0.95**	0.65	0.84	0.82
Our Model - MS-SSIM Loss	0.86	0.94	0.76	0.87	0.86
Our Model - MAE Loss	0.85	0.84	0.86	0.85	0.85
Our Model - MSE Loss	**0.875**	0.83	0.91	**0.88**	**0.88**

Table 2. Comparisons of the proposed model and state-of-the-arts on (EyeQ) dataset [15]

	Accuracy	Sensitivity	Specificity	Precision	F1-Score
MCF-Net Model	0.865	0.946	0.51	0.80	0.75
Our Model - SSIM Loss	0.93	0.94	0.90	0.88	0.90
Our Model - MS-SSIM Loss	0.935	0.93	**0.91**	**0.94**	**0.93**
Our Model - MAE Loss	0.94	0.95	0.88	0.90	0.91
Our Model - MSE Loss	**0.942**	**0.954**	0.89	0.90	0.91

For validating the gradability of retinal fundus image classification performance, we also computed the confusion matrix and the overall classification accuracy on the test set of the two datasets.

This allows more detailed analysis than the mere proportion of high classification rate. As shown in Table 3 , the numbers of true positives (TPs) and true negative (TNs) of the proposed model with the EyePACS dataset are 960 and 765 out of 1999 images, respectively. In the case of the EyeQ dataset, the numbers of TPs and TNs obtained by the proposed model are 12432 and 2867 out of 16249 images, respectively.

Table 3. Two resulting confusion matrices with the test sets of the EyePACS and EyeQ datasets

EyePACS datasets			EyeQ datasets		
n=1999	Predicted NO	Predicted YES	n=16249	Predicted NO	Predicted YES
Actual: NO	TN=765	FP=177	Actual: NO	TN=2867	FP=353
Actual: YES	FN=97	TP=960	Actual: YES	FN=597	TP=12432

Based on the results shown in Tables 1 and 2, we recommended to use the MSE function as the main loss function for calculating the the error between the input image and the reconstructed image. In general, we can say that the proposed model based an autoencoder network yields a promising results on improving RIQA for more accurate ocular diseases classification systems.

5. Conclusions

In this work, we proposed a supervised deep learning model based on an autoencoder network. The autoencoder is able to construct the same input image to correctly learn the visual features of fundus image quality. The model also include a classifier that is fed by the extracted features to classify the quality of a retinal image into gradable and ungradableds. The proposed model can generate more interest in the biomedical community to improve the performance of the RIQA tasks, which plays a critical role in applications such as retinal image segmentation and automatic disease diagnosis. Future work aims to use the developed RIQA model for improving the accuracy of multi-task ocular diseases classification.

Acknowledgement

The Spanish Government partly supported this research through project PID2019-105789RB-I00.

References

[1] G. M. Leggio, C. Bucolo, C. B. M. Platania, S. Salomone, and F. Drago, "Current drug treatments targeting dopamine d3 receptor," *Pharmacology & Therapeutics*, vol. 165, pp. 164–177, 2016.

[2] H. Yu, C. Agurto, S. Barriga, S. C. Nemeth, P. Soliz, and G. Zamora, "Automated image quality evaluation of retinal fundus photographs in diabetic retinopathy screening," in *2012 IEEE Southwest symposium on image analysis and interpretation*, pp. 125–128, IEEE, 2012.

[3] J. Zheng, J. Tian, K. Deng, X. Dai, X. Zhang, and M. Xu, “Salient feature region: a new method for retinal image registration,” *IEEE transactions on information technology in biomedicine*, vol. 15, no. 2, pp. 221–232, 2010.

[4] S. Wang, K. Jin, H. Lu, C. Cheng, J. Ye, and D. Qian, “Human visual system-based fundus image quality assessment of portable fundus camera photographs,” *IEEE transactions on medical imaging*, vol. 35, no. 4, pp. 1046–1055, 2015.

[5] P. Costa, A. Galdran, M. I. Meyer, M. Niemeijer, M. Abràmoff, A. M. Mendonça, and A. Campilho, “End-to-end adversarial retinal image synthesis,” *IEEE transactions on medical imaging*, vol. 37, no. 3, pp. 781–791, 2018.

[6] Y. Wang and S. Shan, “Accurate disease detection quantification of iris based retinal images using random implication image classifier technique,” *Microprocessors and Microsystems*, vol. 80, p. 103350, 2021.

[7] M. Ortega, N. Barreira, J. Novo, M. G. Penedo, A. Pose-Reino, and F. Gómez-Ulla, “Sirius: a web-based system for retinal image analysis,” *International journal of medical informatics*, vol. 79, no. 10, pp. 722–732, 2010.

[8] V. K. Singh, H. A. Rashwan, A. Saleh, F. Akram, M. M. K. Sarker, N. Pandey, and S. Abdulwahab, “Refuge challenge 2018-task 2: Deep optic disc and cup segmentation in fundus images using u-net and multi-scale feature matching networks,” *arXiv preprint arXiv:1807.11433*, 2018.

[9] A. K. Moorthy and A. C. Bovik, “Blind image quality assessment: From natural scene statistics to perceptual quality,” *IEEE transactions on Image Processing*, vol. 20, no. 12, pp. 3350–3364, 2011.

[10] W. Xue, X. Mou, L. Zhang, A. C. Bovik, and X. Feng, “Blind image quality assessment using joint statistics of gradient magnitude and laplacian features,” *IEEE Transactions on Image Processing*, vol. 23, no. 11, pp. 4850–4862, 2014.

[11] T. Brandao and M. P. Queluz, “No-reference quality assessment of h. 264/avc encoded video,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 11, pp. 1437–1447, 2010.

[12] L. Liu, B. Liu, C.-C. Su, H. Huang, and A. C. Bovik, “Binocular spatial activity and reverse saliency driven no-reference stereopair quality assessment,” *Signal Processing: Image Communication*, vol. 58, pp. 287–299, 2017.

[13] F. Shao, W. Lin, S. Wang, G. Jiang, M. Yu, and Q. Dai, “Learning receptive fields and quality lookups for blind quality assessment of stereoscopic images,” *IEEE transactions on cybernetics*, vol. 46, no. 3, pp. 730–743, 2015.

[14] M.-J. Chen, C.-C. Su, D.-K. Kwon, L. K. Cormack, and A. C. Bovik, “Full-reference quality assessment of stereopairs accounting for rivalry,” *Signal Processing: Image Communication*, vol. 28, no. 9, pp. 1143–1155, 2013.

[15] H. Fu, B. Wang, J. Shen, S. Cui, Y. Xu, J. Liu, and L. Shao, “Evaluation of retinal image quality assessment networks in different color-spaces,” in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 48–56, Springer, 2019.

[16] 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, Springer, 2015.

[17] T. Köhler, A. Budai, M. F. Kraus, J. Odstrčilik, G. Michelson, and J. Hornegger, “Automatic no-reference quality assessment for retinal fundus images using vessel segmentation,” in *Proceedings of the 26th IEEE international symposium on computer-based medical systems*, pp. 95–100, IEEE, 2013.

[18] U. Sevik, C. Kose, T. Berber, and H. Erdol, “Identification of suitable fundus images using automated quality assessment methods,” *Journal of biomedical optics*, vol. 19, no. 4, p. 046006, 2014.

[19] R. Pires, H. F. Jelinek, J. Wainer, and A. Rocha, “Retinal image quality analysis for automatic diabetic retinopathy detection,” in *2012 25th SIBGRAPI Conference on Graphics, Patterns and Images*, pp. 229–236, IEEE, 2012.

[20] B. Graham, “Kaggle diabetic retinopathy detection competition report,” *University of Warwick*, 2015.

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

[22] A. Paszke, S. Gross, S. Chintala, and G. Chanan, “Pytorch: Tensors and dynamic neural networks in python with strong gpu acceleration,” 2017.

This page intentionally left blank

Explainability and Argumentation

This page intentionally left blank

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210162

Deep Causal Graphs for Causal Inference, Black-Box Explainability and Fairness

Álvaro PARAFITA and Jordi VITRIÀ
Departament de Matemàtiques i Informàtica,
Universitat de Barcelona, Spain
parafita.alvaro@ub.edu, jordi.vitria@ub.edu

Abstract. Causal Estimation is usually tackled as a two-step process: identification, to transform a causal query into a statistical estimand, and modelling, to compute this estimand by using data. This reliance on the derived statistical estimand makes these methods *ad hoc*, used to answer one and only one query. We present an alternative framework called Deep Causal Graphs: with a single model, it answers any identifiable causal query without compromising on performance, thanks to the use of Normalizing Causal Flows, and outputs complex counterfactual distributions instead of single-point estimations of their expected value. We conclude with applications of the framework to Machine Learning Explainability and Fairness.

Keywords. causality, counterfactual, causal graph, flow, explainability, fairness

1. Introduction

Artificial Intelligence requires causal knowledge to determine how its actions or predictions may affect or be affected by the outside world. As an example, the association between ice-cream sales and shark attacks might seem causal, when it is in fact due to a latent confounder between them, summer, which makes it spurious. Defining the causal graph that specifies each dependency is just the first step, as we also need an estimation engine to compute the result of causal queries (*i.e.*, ”what is the effect of administering the treatment on a patient?”). Causality is also key to black-box Explainability and Fairness, as we can explain the effect of certain input features as interventional effects, or a prediction’s fairness on an individual as the counterfactual effect of protected features, such as gender or race.

This relies on our ability to estimate causal queries (*i.e.*, $p(\text{salary} \mid do(\text{education} = \text{undergraduate}))$, meaning, the probability of having a certain salary were we forced to get an undergraduate degree). Estimation usually entails a preprocessing step, *identification*, which transforms our causal query into a statistical estimand that can be estimated with observational data. The problem with this framework is that its models are *ad hoc* to the causal query at hand: were we to ask a new question, we would need to train an additional model. Moreover, most are designed to only estimate the expected value of the target distribution, which does not account for multimodality, skewness, etc.

The contributions of this paper are twofold. Firstly, we introduce **Deep Causal Graphs** (DCG), a sampling-based framework with which we can answer any *identifi-*

able causal queries in a graph with the same trained model. Secondly, we propose **Normalizing Causal Flows**, an implementation of this specification based on Conditional Normalizing Flows, which allows us to model complex continuous distributions that can be integrated in our DCGs. Finally, we provide experiments that showcase the quality of our model's estimations and the applicability of these techniques to the fields of **Machine Learning Explainability and Fairness**. Additionally, we provide a PyTorch library which includes all DCG implementations in this paper and functionality for running experiments on the showcased applications. The code can be found in a public repository (github.com/aparafita/ccia2021supp) along with the supplementary material.

2. Related Work

Causal Theory and their applications have mainly been studied from two perspectives: Potential Outcomes [1] and Causal Graphs [2]. Our work focuses on the latter, specially regarding Structural Equation Models (SEM). Nevertheless, the Potential Outcomes perspective is compatible with the Causal Graph literature, and we incorporate some of the findings in [3] in our work.

Deep Causal Graphs model the aforementioned SEMs, but also leveraging the expressive power of deep neural networks. From this point of view, they are directly related to two previous works. CausalGAN [4] represented each random variable as a neural network with their parents' values as the input. Our previous work, Distributional Causal Nodes [5], extended this idea by assuming a known parametric probability distribution for each node. Our current approach subsumes the latter by modelling arbitrary distributions using Normalizing Flows [6], instead of known parametric families. Independent work in [7] also proposes Normalizing Flows for Counterfactual Estimation applied to MRI scans.

In terms of applications, Deep Causal Graphs are specially suited to counterfactual explanations. Counterfactual reasoning has been proposed as an important ingredient for explainability and fairness analysis (*i.e.*, [8,9,10]), but in most of these frameworks, counterfactuals are understood as samples with minimal alterations in the input that change a black-box prediction. This definition does not take into account the causal effects of these alterations on the rest of the variables, therefore providing non-actionable explanations. Our model does work with intervened distributions, circumventing this issue. Additionally, it allows a practical implementation of Counterfactual Fairness [11], which measures the effects of interventions of protected variables on the target variable.

3. Background

We define a **Structural Equation Model** (SEM) as the tuple $M = (V, E, U, P(E), P(U), F)$:

1. $V = \{V_1, \dots, V_K\}$, the measured random variables.
2. $E = \{E_1, \dots, E_K\}$, the exogenous noise variables, one for each V_k.
3. $\emptyset \subseteq U \subseteq \{U_{\{k,l\}}\}_{k,l=1..K,k<l}$, the latent (non-measured) confounder variables $U_{\{k,l\}}$ that explain unmeasured common causes between V_k and V_l.
4. $P(E)$ and $P(U)$, the prior distributions for all non-measured variables.

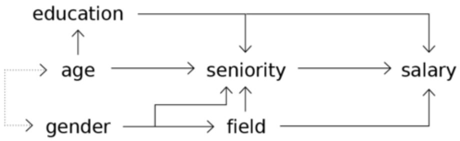


Figure 1. *Salary* dataset causal graph. Bidirectional arrows represent latent confounders.

5. $F = \{f_k = f_k(Pa_k, U_{\{k,.\}}, E_k)\}_{k=1..K}$, the functional relationships[1] $V_k = f_k(.)$ between a node V_k, its measured parent set $Pa_k \subsetneq V$, its parent latent variables $U_{\{k,.\}}$ (if any) and its corresponding E_k.

F implicitly defines a directed graph $G_M = (\mathscr{V} := V \cup E \cup U, \mathscr{E})$ where $\mathscr{E}$, its edges, are defined by all input-output relationships in F: $\mathscr{E} = \bigcup_{k=1..K}\{(V,V_k) \mid V \in Pa_k\} \cup \{(U,V_k) \mid U \in U_{\{k,.\}}\} \cup \{(E_k,V_k)\}$. Any directed edge represents a causal dependency between source/cause and target/effect. A common assumption is that G_M is a Directed Acyclic Graph (DAG). See figure 1 for an example.

3.1. Interventions and Counterfactuals

There are two important operations to consider in Causal Theory. Firstly, the **intervention**, that alters the distribution represented by the model. Specifically, a constant-value intervention, normally represented by $do(X = x)$, means replacing the generative function $X = f_X(.)$ by an assignment $X = x$. This constant value x comes predefined by the intervention and does not depend on the parents of X. Therefore, the intervened SEM replaces f_X by this assignment and the corresponding intervened graph is the subgraph where all edges pointing at X are removed. This subgraph encodes a different distribution, the **intervened distribution**.

Secondly, the **counterfactual**. Given a certain observational sample e' of $E' \subseteq V$ and an intervention $do(X = x)$, a counterfactual is the result of an hypothetical experiment in the past, what would have happened to the values of variables $Y \subseteq V$ had we intervened on X by assigning value x. Counterfactual expressions are of the form $p(Y_x \mid e')$, with Y_x the counterfactual variables under intervention $do(X = x)$. Pearl [2] defines counterfactuals as a three-step process: **abduction**, compute the posterior distribution[2] of the latent variables E and U conditioned on evidence e', $p(E, U \mid e')$; **intervention**, apply the desired intervention $do(X = x)$; **prediction**, compute the required prediction in the intervened, counterfactual model $\widehat{M}$ defined by the abducted priors and the modified set of functions $\widehat{F}$, where f_X has been replaced by $X := x$.

3.2. Identifiability and Structural Causal Models

An essential matter in causal inference is that of a query's **identifiability**. Given a causal query for a certain generative process described by a graph G and a distribution $P(V)$, we say it is identifiable if we can derive an statistical estimand (only using observational terms) for this query using the rules of do-calculus [2]. Two SCMs both representing $P(V)$ with the same causal structure G will output the same result for such a causal query. In

[1] Note that, although f_k is deterministic, the effect of E_k makes it stochastic *w.r.t.* Pa_k, $U_{\{k,.\}}$.

[2] We refer to these new distributions as the *abducted priors*.

other words, if we model an SCM M that represents the same distribution while following the same causal structure as the generative process, it is guaranteed to generate the correct estimations for any identifiable causal query (including counterfactuals), **no matter if the functions in $\boldsymbol{F_M}$ nor the latent priors $\boldsymbol{P(E), P(U)}$ do not match the real generative process.**

In order to determine identifiability, there are automatized solutions for different types of queries: purely interventional queries ($i.e.$, $p(y \mid do(X = x))$) [12], post-intervention conditional queries ($i.e.$, $p(y \mid do(X = x), z)$) [13], and counterfactual queries in an arbitrary number of parallel, counterfactual worlds [14]. As such, one can automatically determine if our techniques can be used for the estimation of a certain causal query. If not, alternatives such as instrumental variables, more restrictive parametric assumptions (when they apply) or randomized experiments (even for the counterfactual case, using the latter referenced work) could circumvent this issue.

4. Method

4.1. Deep Causal Graph

A Deep Causal Graph (DCG) is an abstract specification of the required functionality for a Deep Neural Network to work with causal queries. The only assumption is positivity: $p(v) > 0$ for all v in the domain of V. In other words, all possible configurations of the graph's variables are possible, no matter how unlikely. DCGs model the SEM described in section 3: given a SEM $M = (V, E, U, P(E), P(U), F)$, we represent each random variable in V as a subcomponent called the Deep Causal Unit (DCU), with three operations:

- $\boldsymbol{sample}(\textbf{parents})$: sample a new realization of the variable, given its parents values.
- $\boldsymbol{loglk}(\textbf{sample}, \textbf{parents})$: compute the log-likelihood of the sample, given its parents values. This operation must be **differentiable** *w.r.t.* the DCU's parameters.
- $\boldsymbol{abduct}(\textbf{sample}, \textbf{parents})$: sample from the abducted noise $(E_k \mid X_k, Pa_k, U_{\{k,.\}})$.

Given these three operations for each node, we can perform estimation across the graph. Assuming nodes in topological order, sampling consists of iteratively applying each node's ***sample*** operation. Any latent variables in E and U can be sampled from their priors $P(E)$ and $P(U)$ directly. Interventions are performed by replacing the *sample* operation with an assignment to the intervened value. However, computing **log-likelihoods** is more nuanced. Given a sample v of variables $\emptyset \subsetneq V' \subseteq V$ (some may be missing), let us define $Z := U \cup \{E_X \in E \mid X \in V \setminus V'\}$. If Z is empty, no variable is missing, which allows us to apply the conditional independencies entailed by the graph G_M (*d-separability*, [2]): $\log p(v) = \sum_{k=1..K} \log p(v_k \mid pa_k)$. If Z is not empty, given a sample $z \sim P(Z)$, we can generate a value for each missing variable in V' deterministically. Then, $\log p(v) = \log \mathbb{E}_Z \left[p(v \mid Z) \right] \approx \log \frac{1}{N} \sum_{i=1}^{N} \exp \sum_{k; V_k \in V'} \log p(v_k \mid pa_{k,i}, u_{\{k,.\},i})$, with N i.i.d samples $z_i \sim P(Z)$, from which we can obtain every $u_{\{k,.\},i}$ and every $pa_{k,i}$. In this case, we only compute the log-likelihood of the variables in V', not every measurable variable in the graph. Additionally, we employ the log-sum-exp trick for numerical stability.

The requirement for the DCU's ***loglk*** operation to be differentiable entails that the graph's ***loglk*** is also differentiable. As a result, we can train all nodes simultaneously by Maximum Likelihood Estimation: assuming i.i.d. data, we can train with Stochastic

Gradient Descent by maximizing the average log-likelihood of the dataset. Be warned that, despite being able to compute log-likelihoods of incomplete samples, one should not train with missing data blindly, as the missingness mechanism could induce biases in $P(V)$. An identification of feasible scenarios for training with missing data is left for future work.

The final operation is ***counterfactual***, which, given evidence e of variables $\emptyset \subsetneq E \subseteq V$ and an intervention $do(X=x)$, generates $v_x = (v_x^{(i)})_{i=1}^N \sim P(V_x \mid e)$, N counterfactual samples. To do that, we follow the three-step process defined before: abduction (call each node's *abduct* operation), intervention (apply $do(X=x)$) and prediction (sample each counterfactual node V_x). If there were missing values in our evidence ($E \subsetneq V$) or latent confounders in the graph ($U \neq \emptyset$), we would not have access to every parent's value, which is required by the *abduct* operation. In that case, we use **importance sampling**, which provides us with samples $v_x = (v_x^{(i)})_{i=1}^N$ and corresponding weights $w = (w^{(i)})_{i=1}^N$. These weighted samples allow us to 1) compute counterfactual queries from this distribution, using weighted averages, and 2) study the counterfactual distribution directly, by generating a weighted Bootstrap subsample of (v_x, w). The generation of these samples and weights, along with practical considerations on the implementation of these techniques, is left for the supplementary material due to space restrictions.

4.2. Deep Causal Unit

This subsection discusses two possible implementations of the DCU. Distributional Causal Nodes (DCN) [5] assign a parametric probability distribution to every node V_k (*i.e.*, Gaussians, Exponentials or Categoricals) with parameters Θ_k and model their distribution by defining a neural network f_k that takes the node's parents as input and computes the distribution's parameters Θ_k as the output ($\Theta_k = f(Pa_k, U_{\{k,\cdot\}})$). With this we can perform all three DCU operations: 1) *sample*, by using E_k as an independent noise signal with a reparametrization trick [15] for the assumed distribution; 2) *loglk*, by using the density of the assumed parametric distribution; 3) *abduction*, by inverting the reparametrization formula. This inversion might not always be possible, in which case we could use rejection sampling. In particular, Bernoulli and Categorical distributions can be abducted when using the Gumbel-softmax trick [16] as the sampling step. There are, however, two disadvantages to DCNs. On the one hand, users need to specify a well-matched distribution for each node in the graph, which can be costly on graphs with many variables. On the other hand, standard distributions might not be sufficient to properly adjust complex datasets.

To avoid these two issues, for continuous DCUs, we propose instead **Normalizing Causal Flows** (NCF). A Conditional Normalizing Flow models probability distributions $P(X \mid Z)$ by transforming a continuous random variable X into a base distribution E_X (usually a standard normal distribution) of the same dimension as X, with a parametric function f_Z invertible *w.r.t.* X whose parameters depend on the conditioning variable Z. Here Z consists of the parents values of node X, Pa_X and $U_{\{X,\cdot\}}$; therefore, f allows us to model $P(X \mid Pa_X, U_{\{X,\cdot\}})$, which is precisely what we need in our DCU. Moreover, this invertible function between X and E_X is guaranteed to exist under certain regularizing conditions [6]. The main advantage of this setup is that we can compute $P_X(x \mid z) = P_{E_X}(f_z(x)) \cdot |\det J_{f_z}(x)|$, where $J_{f_z}(x)$ is the Jacobian of f_z on x. Normally, we model this function using a conditioner-transformer architecture (see [6] for more details) with the

Table 1. Metrics on IHDP and Jobs datasets, for the training and test sets. Lower is better.

	IHDP				JOBS			
	$\sqrt{e_{PEHE}}$		e_{ATE}		R_{POL}		e_{ATT}	
	TRAIN	TEST	TRAIN	TEST	TRAIN	TEST	TRAIN	TEST
LR_1	5.8 ± .3	5.8 ± .3	.73 ± .04	.94 ± .06	.22 ± .00	.23 ± .02	**.01 ± .00**	.08 ± .04
LR_2	2.4 ± .1	2.5 ± .1	**.14 ± .01**	.31 ± .02	.21 ± .00	.24 ± .01	.01 ± .01	.08 ± .03
BNN	2.2 ± .1	2.1 ± .1	.37 ± .03	.42 ± .03	.20 ± .01	.24 ± .02	.04 ± .01	.09 ± .04
TAR	.88 ± .02	.95 ± .02	.26 ± .01	.28 ± .01	.17 ± .01	.21 ± .01	.05 ± .02	.11 ± .04
CFR	**.71 ± .02**	**.76 ± .02**	.25 ± .01	.27 ± .01	**.17 ± .01**	**.21 ± .01**	.04 ± .01	.09 ± .03
DCG	1.0 ± .05	1.2 ± .09	.20 ± .01	**.25 ± .02**	.22 ± .01	.24 ± .05	.05 ± .02	**.04 ± .01**
DCG*	.94 ± .04	1.0 ± .06	.20 ± .01	**.23 ± .02**	**.11 ± .02**	**.12 ± .04**	.05 ± .01	**.05 ± .01**

conditioner depending on the conditioning input z, while the transformer is an invertible neural network with certain architectural constraints to make this inversion possible and its determinant tractable. As a result, our flow f is capable of: 1) ***sampling*** from $P(X \mid pa_X, u_{\{X,\cdot\}})$ by sampling an $\varepsilon \sim p(E_X)$ and transforming it back to X with $x = f^{-1}_{pa_X, u_{\{X,\cdot\}}}(\varepsilon)$; 2) computing the ***log-likelihood*** of a realization x as described before; 3) computing the $\varepsilon_x \sim p(E_X)$ such that $f_{pa_X, u_{\{X,\cdot\}}}(x) = \varepsilon_x$ (***abduction***).

In summary, any type of conditional Normalizing Flow can be used in a graph to model complex continuous random variables thanks to their high expressiveness, while also avoiding DCN's node-wise distributional assumptions.

5. Experiments

To evaluate our technique, we will study two benchmark datasets: the Infant Health and Development Program (IHDP dataset, [17]) and the study in [18] about National Supported Work (Jobs dataset). We follow the setup and results from [3], focusing on four metrics, two for each dataset, respectively: *estimated Precision on Estimation of Heterogeneous Effect* (e_{PEHE}), *error in Average Treatment Effect* (e_{ATE}), *error in Average Treatment effect for the Treated* (e_{ATT}) and *Policy Risk* ($R_{pol}(\pi_f)$). Both datasets contain many replications of its samples, so that it is possible to obtain a confidence interval of each metric by training a model with each replication. For fairness in comparison, we only train the DCU of the target variable, as the rest of the models do. Details about the experiment setup, model implementation and the actual code can be found in the supplementary material. For reference, training each of our models in a GPU takes on average 16 seconds on IHDP and 28 seconds on Jobs.

Additionally to the base DCG estimation, we consider a variant that computes the counterfactual outcome also using y_f, the factual outcome. When analyzing alternative outcomes *ex post facto* (such as discussing a loan rejection in a bank, and asking for the alternative outcome had we changed a certain variable), we do have access to the factual outcome, and using it results in better estimation of the counterfactual, as we will see. In contrast with DCGs, some of the methods with which we compare do not provide this functionality, since they do not consider the required abduction step in the counterfactual process. We include this extra case to compare the performance of our method with the rest of the benchmark, for cases where such an input is available.

Results can be found in table 1 for: Linear Regression (LR_1), separate LR for each treatment (LR_2), Balancing Neural Network (BNN, [19]), as well as Treatment Agnostic Representation Network (TAR, TARNet) and its variant with balancing regularization, Counterfactual Regression with Wasserstein distance (CFR) (both from [3]). We include our model as DCG and its variant using y_f as DCG*. DCG achieves comparable results with TARNet in $\sqrt{e_{PEHE}}$ in both sets. For e_{ATE}, DCG is the best model (only surpassed by LR$_2$ in training, but not in test). In the Jobs dataset, DCG achieves comparable results to BNN on the Policy Risk metric, far from the results with TARNet, but not in the case of DCG*, with which we surpass every other method. Finally, e_{ATT} in training seems to be significantly worse than far-simpler methods, but not in test, where DCG surpasses every other model in both its variants.

TARNet and CFR obtain better results in some instances (PEHE and Policy Risk), even though our network structure also uses their bi-headed networks and they are mostly equivalent (except for CFR's balancing regularization). This difference in results can be attributed to the fact that DCGs learn the distribution of these outcomes, in contrast with the other networks, which just return the expected value. This broader objective might hinder the accuracy of the expectation estimation, but it can be worthwhile nevertheless since we can analyze the distribution afterwards, looking for multi-modality, high skewness, etc., as we will see in the next section. Hence, considering this additional feature, we find that our method is comparable to the other two.

6. Applications

The objective of this section is to apply DCGs to Explainability and Fairness of black-box predictors. We created a synthetic gender wage gap dataset containing several mechanisms that create bias against women in terms of salary (see figure 1). *Gender* affects the choice of *field*, due to societal pressures, and *seniority*, due to management bias in promoting men and women. Additionally, we consider a plausible form of selection bias resulting from parents who leave work to take care of their children, with more incidence on women. This process biases the data so that people who are still working while being older (hence more likely to have children) are mostly men. We model this bias with a latent confounder between *gender* and *age*. The final dataset contains 6,479 samples.

We train a DCG with NCFs and Bernoulli-DCNs (depending on each node having continuous or Bernoulli distributions, respectively) with the same configurations as in the experiments. Additionally, we train a Multi-Layer Perceptron (MLP) regressor of *salary*, which will be our black-box. The objective is twofold: explain the predictions of the MLP with a causal perspective (how would our prediction be had we changed any variable) and measure and reduce the Counterfactual Unfairness of these predictions *w.r.t.* gender.

6.1. Machine Learning Explainability

If we want to estimate the effect of gender on the salary prediction, we compute $\mathbb{E}[\widehat{S} \mid do(G = male)] - \mathbb{E}[\widehat{S} \mid do(G = female)]$. For that we sample using each intervention, generating values for every variable in the graph, and then use them as the input of the MLP, obtaining $(\widehat{s}_i)_{i=1..n}$. The averages of these samples approximate each expectation, which allows us to compute the desired effect: \$3,549. In contrast, were we to estimate the observational query $\mathbb{E}[\widehat{S} \mid G = male] - \mathbb{E}[\widehat{S} \mid G = female]$, we would get \$4,878.

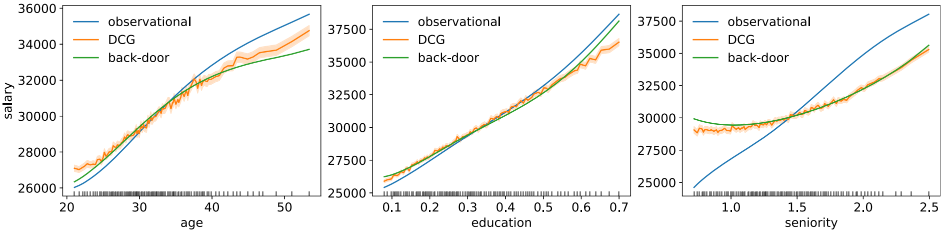


Figure 2. Observational (blue) and interventional effect of three continuous variables on *salary*. The latter is estimated with DCGs (orange) and with the Back-Door Adjustment formula (green) for comparison.

Next, we will study the effects of several continuous variables on salary. Most causal estimation methods argue that their techniques can be easily extended to the continuous case, but many fail to provide examples of this. Figure 2 contains the effect of *age*, *education* and *seniority* on *salary*. In orange, we plot the expected value for each intervention (each x value) with a 95% confidence interval (generated by sampling 1,000 points per intervention and aggregating them). In blue, we include the observational effect ($p(salary \mid X)$) for every variable as a reference to compare between interventional and observational effects. In green, we estimate each interventional effect with an estimand derived from the Back-Door Adjustment formula. Both the blue and green curves are fitted using a 5-degree polynomial basis with Linear Regression. Both interventional curves should and do match, with the only exception of the outlying values (notice the ticks on the x-axis), which is to be expected. Nevertheless, note that with the usual method, we need a different *ad hoc* model for each new query, in contrast with DCGs, which are trained once and can be used on all (identifiable) queries. No matter how complex the estimand might be, DCGs will always be trained equally (using MLE) and will estimate any identifiable query using the same procedures.

Finally we study counterfactuals, to explain predictions on a particular individual. Given evidence e, what would the predicted salary $\widehat{S}$ be had we intervened with $do(X=x)$. This kind of query is normally answered as an expectation; however, since we can sample from the counterfactual distribution, we will plot its density instead. Figure 3 shows counterfactual distributions based on evidence $\{$woman, 30 years old, \$30,000 annual salary$\}$. On the left, we intervene by changing her gender and find a bi-modal distribution. The green vertical line represents the average counterfactual salary; note that this average would not inform us of the two modes and provides an unlikely counterfactual outcome. On the right, we intervene on 50 values of age (50 equidistant quantiles of age from 2.5% to 97.5%) and plot the corresponding densities, with colour representing each value of age. In orange we can see the original salary as the factual outcome. As age increases, the *salary* distribution moves to higher values, as one would expect.

6.2. Counterfactual Fairness

To conclude, we will study how to mitigate the unfairness of the MLP regressor. Before, we saw that *gender* did have an effect on *salary* that created a significant gender wage gap. We can also measure it on a specific individual: on sample 4,746, a woman with predicted salary \$34,551, the average counterfactual salary had she been a man results in \$43,077, a difference of \$8,526 (much higher than the average for the population, in her case).

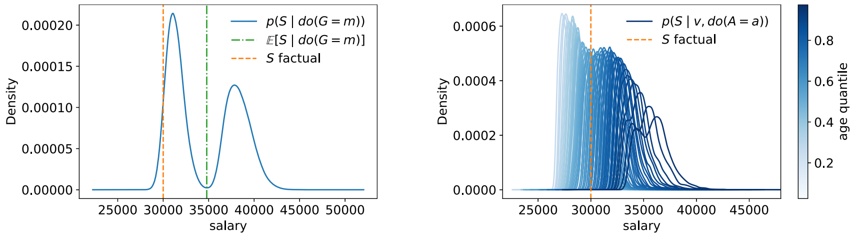


Figure 3. Counterfactual *salary* density curves with interventions on *gender* (left) and *age* (right). Both plots are based on evidence $v := \{\text{woman, 30 years old, \$30,000 annual salary}\}$.

Next, we will train a fairer predictor through Counterfactual Fairness [11], a measure of bias between observational and counterfactual samples. Let us define Counterfactual Unfairness of degree k as $CU_k := \mathbb{E}_V\left[\mathbb{E}_{E,U|V}\left[|Y_{1-x}(E,U) - Y(E,U)|^k\right]\right]$ where X are the intervened (protected) variables, Y the observed target variable and Y_{1-x} the counterfactual target variable. The CU_1 of *salary* (the average unsigned difference between counterfactual and real values) is \$3,883, which shows a significant bias in the model. Note that we can train a differentiable model adding CU_2 as a regularization term; the resulting model (with regularization weight of 10) has a CU_1 of \$234, making it, indeed, fairer. If we test this new regressor on the previous woman, the predicted and counterfactual *salary* estimations become \$38,960 and \$39,379, respectively, with a bias of \$419.

Note that a drop in performance is to be expected, as we are modelling a non-biased distribution, not the original. In this case, it might be preferable to ensure that the resulting ordering (inside each group) matches the one in the dataset. The original predictor, which had an R^2 score of 98%, has a Spearman correlation (*w.r.t.* the original data) of 99% for both genders, while the fair model's correlation is of 90% and 89% for women and men, respectively, even though its R^2 decreases to 68%. Therefore, the choice of metric is essential, since the original data might not reflect the fair world we want to model.

7. Conclusions

We propose Deep Causal Graphs, a flexible and powerful Causal Estimation framework that allows answering any identifiable causal query in a graph by training only once with Maximum Likelihood Estimation. Its fitting capabilities are guaranteed by the use of any Conditional Normalizing Flow and/or Distributional Causal Node. Instead of returning point estimates of causal queries, by its sampling-based nature, it can output complex distributions that result in richer objects of study. Additionally, we showcase many applications of Causal Estimation, powered by DCGs, to the fields of black-box Explainability and Counterfactual Fairness. Further work on alternative implementations of Deep Causal Units or on training with other forms of data (*i.e.*, multivariate node variables, like images, or non-i.i.d. data, like time series) could further extend the applicability of the framework. A PyTorch library is also included for further testing and extensions of the technique.

Acknowledgements

This work has been partially funded by projects RTI2018-095232-B-C21 (MINECO/FEDER), 2017.SGR.1742 (Generalitat de Catalunya) and Universitat de Barcelona's APIF program.

References

[1] Rubin DB. Causal inference using potential outcomes: Design, modeling, decisions. Journal of the American Statistical Association. 2005;100(469):322–331.

[2] Pearl J. Causality. Cambridge university press; 2009.

[3] Shalit U, Johansson FD, Sontag D. Estimating individual treatment effect: generalization bounds and algorithms. In: International Conference on Machine Learning. PMLR; 2017. p. 3076–3085.

[4] Kocaoglu M, Snyder C, Dimakis AG, Vishwanath S. CausalGAN: Learning Causal Implicit Generative Models with Adversarial Training. In: International Conference on Learning Representations (ICLR); 2018. .

[5] Parafita Á, Vitrià J. Explaining Visual Models by Causal Attribution. In: 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW). IEEE; 2019. p. 4167–4175.

[6] Papamakarios G, Nalisnick E, Rezende DJ, Mohamed S, Lakshminarayanan B. Normalizing Flows for Probabilistic Modeling and Inference. arXiv:191202762 [statML]. 2019.

[7] Pawlowski N, Coelho de Castro D, Glocker B. Deep Structural Causal Models for Tractable Counterfactual Inference. Advances in Neural Information Processing Systems. 2020;33.

[8] Wachter S, Mittelstadt B, Russell C. Counterfactual Explanations without Opening the Black Box: Automated Decisions and the GPDR. Harvard Journal of Law & Technology (Harvard JOLT). 2017;31:841.

[9] Goyal Y, Wu Z, Ernst J, Batra D, Parikh D, Lee S. Counterfactual Visual Explanations. In: Proceedings of the 36th International Conference on Machine Learning. vol. 97; 2019. p. 2376–2384.

[10] Mothilal RK, Sharma A, Tan C. Explaining Machine Learning Classifiers through Diverse Counterfactual Explanations. In: Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency; 2020. p. 607617.

[11] Kusner MJ, Loftus J, Russell C, Silva R. Counterfactual fairness. In: Advances in Neural Information Processing Systems; 2017. p. 4066–4076.

[12] Shpitser I, Pearl J. Identification of joint interventional distributions in recursive semi-Markovian causal models. In: Proceedings of the National Conference on Artificial Intelligence. vol. 21; 2006. p. 1219.

[13] Shpitser I, Pearl J. Identification of conditional interventional distributions. In: 22nd Conference on Uncertainty in Artificial Intelligence, UAI 2006; 2006. p. 437–444.

[14] Shpitser I, Pearl J. What counterfactuals can be tested. In: 23rd Conference on Uncertainty in Artificial Intelligence, UAI 2007; 2007. p. 352–359.

[15] Kingma DP, Welling M. Auto-Encoding Variational Bayes. In: 2nd International Conference on Learning Representations, ICLR 2014. Banff, AB, Canada; 2014. .

[16] Jang E, Gu S, Poole B. Categorical Reparameterization with Gumbel-Softmax. In: 5th International Conference on Learning Representations, ICLR 2017. Toulon, France; 2017. .

[17] Hill JL. Bayesian nonparametric modeling for causal inference. Journal of Computational and Graphical Statistics. 2011;20(1):217–240.

[18] LaLonde RJ. Evaluating the econometric evaluations of training programs with experimental data. The American economic review. 1986:604–620.

[19] Johansson F, Shalit U, Sontag D. Learning representations for counterfactual inference. In: International conference on machine learning. PMLR; 2016. p. 3020–3029.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210163

Improving On-Line Debates by Aggregating Citizen Support*

Maite LOPEZ-SANCHEZ [a,1], Marc SERRAMIA [b,a], and
Juan A. RODRÍGUEZ-AGUILAR [b]
[a] *Universitat de Barcelona (UB)*
[b] *Institut d'Investigació en Intel·ligència Artificial (IIIA-CSIC)*

Abstract. Currently, Digital Democracy is gaining momentum thanks to online participation platforms, which have emerged as innovative tools that enable citizens to participate in decision making processes. Through these tools, participants can issue proposals and engage into debates by both stating arguments in favour or against and/or by supporting other people's arguments. In this paper we propose a new support aggregation method derived from the combination of two complementary aggregation methods previously introduced. Additionally, we propose a resilience metric for measuring the quality of the aggregated opinion. We apply our contributions to debates conducted in the Decidim participatory platform.

Keywords. AI applied to participation, information fusion, debate quality.

1. Introduction

Most currently established democracies follow a *representative democracy* model where people periodically elect persons to represent them. However, it is often the case that people feel disconnected from their representatives, who retain the power during the inter-election periods. Against this, participatory portals are designed to bridge the gap between citizens and their governments. They foster participation in the public sphere by supporting the co-creation of proposals, public deliberation or joint decision-making. This supposes a transformation of the traditional model of representative democracy into a model of participatory democracy [19,7] where citizens become involved in decisions about some of the issues that most affect them, that is, the public interest [17]. This transition, in addition to mitigating the disaffection that a part of the citizenry feels towards politics, can lead to a substantial improvement in the acceptance of the decisions taken.

Participatory portals can also help to face challenges and crises (e.g., sustainable development goals, climate change, or pandemics) in a more cohesive way by enabling informed and reasoned decision making processes and by helping to mitigate the increasing polarisation of societies. In fact, debate has been proposed as an antidote to polarisation

[1] Corresponding Author: Maite Lopez-Sanchez; E-mail:maite_lopez@ub.edu

* Research supported by the Laboratori d'Innovació Democràtica and funded by the Ajuntament de Barcelona, the Fundació Solidaritat UB and projects 2017 SGR 172, 2017 SGR and 341, and AI4EU (H2020-825619), Crowd4SDG (H2020-872944), CI-SUSTAIN (PID2019-104156GB-I00), COREDEM (H2020-785907), nanoMOOC (COMRDI18-1-0010-02), MISMIS (PGC2018-096212-B-C33).

Citizen participation
facing covid-19
Le jour d'après
BARCELONA
DES DE CASA
Hackovid
Imagen adaptada de decideix Gràcia https://www.youtube.com/watch?v=jGQqWSFqzFQ (d.decidim)
decidim

Figure 1. Different initiatives to face COVID-19 pandemic enabled by the Decidim platform.

from some social actors such as the initiative of Spaceship Media [10], which emerged in the United States in the context of the political fracture produced in the Trump era. They argue that dialogue from difference is essential for the proper functioning of democracy.

We find numerous platforms for citizen participation. Among these, it may be worth mentioning: the French initiative Purpoz (previous “Parlement et Citoyens”[8]), where we can nowadays find debates on the climate bill or vaccination; the Petitions UK platform [13] in the United Kingdom, where 441 petitions got a response from the Government and 71 petitions were actually debated in the House of Commons by May 2021; or Rousseau in Italy [15] named after the theorist *par excellence* of participation [12]. Interestingly, some of the participatory democracy platforms –such as Decidim [6] or Consul [5]– emerged from popular movements, are open source, and are being used by local governments (e.g. the Generalitat de Catalunya [11]) and numerous municipalities such as Madrid, Reykjavik [1], New York, Buenos Aires or Barcelona [4]. Indeed, their use is not limited to public administrations, since organizations such as universities (e.g., Universidade da Coruña) or cooperatives (e.g., Som energia) are also using them to make their governance model more participatory.

In fact, although the pandemic has further promoted the digital divide among the most disadvantaged population, it has also turned out to be an indisputable catalyst for the digitalization of most of the population. In this manner, it poses a unique opportunity for participatory tools to demonstrate that the required dedication effort is worthy because people can feel part of the community and its decision process. For instance, Figure 1 illustrates that Decidim [6] was readily used in different initiatives facing the COVID-19 pandemic. If this opportunity is seized, the gain in transparency, traceability and accountability, as well as inclusion (both social and gender inclusion, since we need to transfer the role of women that has traditionally been relegated to the domestic sphere, to open it to participation in public spaces) represents a qualitative leap forward in our current democratic models.

Deliberation is a key mechanism to guarantee the quality of the decisions made. Indeed, debates in participatory platforms can facilitate informed and reasoned decision making when including the advantages and disadvantages of each proposal. However, the further the discussion progresses (or the larger the number of participants), the more complex to follow the argumentation. That is why it is vitally important to structure these comments in such a way that the arguments that are for and against the proposal are clearly distinguished. In doing so, if participants then express their opinion about these arguments, then information aggregation operators can be applied to combine the different opinions into a single value that represents a joint support to the proposal.

This paper builds upon two aggregation methods –PAM [14] and TODF [9]– that compute this support (or joint opinion) in a more sophisticated way than the typical majorities, which suffer from a certain tendency to centralization. On the one hand, PAM [14] can consider aspects such as the number and importance of the expressed opinions to discard weak arguments that may hinder the resulting aggregated opinion, or even identify cases in which there is insufficient information to correctly evaluate a proposal. On the other hand, TODF [9] exploits the debate structure and guarantees properties –such as anonymity or non-authoritarianism– that have typically been considered within Social Choice Theory. We propose a hybrid aggregation method that inherits the advantages of both PAM and TODF methods and is tailored for the Decidim platform. We analyse these alternative methods by comparing the resulting aggregated support for some instances of real debates from Decidim Barcelona [4].

Furthermore, we propose a resilience measure to assess the quality of the decision taken w.r.t the proposal (i.e., if it is accepted or rejected). We define the resilience by considering the modifications that would be necessary to reverse the decision. Hereby, the more changes needed, the more resilient the decision.

We structure the paper as follows. Next section briefly introduces PAM and TODF aggregation methods so to provide the basis for our hybrid method, which is discussed in Section 3. Subsequent Section 4 proposes the resilience measure and illustrates its usage for some proposals from Decidim Barcelona. Finally, Section 5 concludes the paper and discusses possible future paths for research.

2. Background

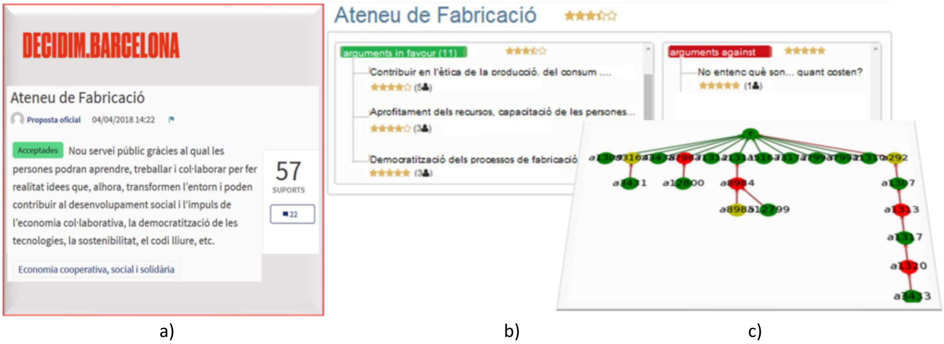
This section is devoted to briefly introduce the two opinion aggregation methods: the *Proposal Argument Map* (PAM) and the *Target oriented discussion framework (TODF)*.

For illustration purposes, Figure 2 a) shows proposal[2] 50 from Decidim Barcelona [4]. It received 57 direct users' supports and aroused a debate consisting of 22 (in favor, against and neutral) arguments. Figure 2 b) illustrates how PAM groups arguments in favour and against, indicates how many citizens expressed their opinions over each argument, and represents the aggregated opinions using a 5-star scale. Figure 2 c) depicts the overall argument structure in the debate of proposal 50 (τ) as represented by TODF. Circles linked with green arrows represent arguments in favour, whereas arguments against are linked with red arrows. TODF signals accepted, undecidible, and rejected arguments (as computed by it) with green, yellow, and red colours respectively.

2.1. Proposal Argument Map (PAM)

Briefly, a proposal argument map (PAM) is a structure representing the content of a debate in terms of those arguments in favour and those arguments against a given proposal. Formally, Rodriguez-Aguilar et al. [14] define it as:

[2] Proposal 50 (Pla d'Actuació Municipal 2016-2019): https://decidim.barcelona/pam/proposals/ateneu-de-fabricacio, https://github.com/elaragon/metadecidim/blob/master/proposals/00050.json



DECIDIM.BARCELONA

Ateneu de Fabricació

Proposta oficial 04/04/2018 14:22

Acceptades Nou servei públic gràcies al qual les persones podran aprendre, treballar i col·laborar per fer realitat idees que, alhora, transformen l'entorn i poden contribuir al desenvolupament social i l'impuls de l'economia col·laborativa, la democratització de les tecnologies, la sostenibilitat, el codi lliure, etc.

57 SUPORTS

22

Economia cooperativa, social i solidària

Ateneu de Fabricació ★★★☆☆

arguments in favour (11) ★★★☆☆

Contribuir en l'ètica de la producció, del consum ....
★★★★☆ (5)

Aprofitament dels recursos, capacitació de les persones...
★★★★☆ (3)

Democratització dels processos de fabricació
★★★★★ (3)

arguments against ★★★★★

No entenc què son... quant costen?
★★★★★ (1)

Figure 2. Proposal 50 as in a) Decidim (just general description shown), b) PAM, c) TODF.

Definition 1. *A Proposal Argument Map (PAM) is a triple $\langle p, A_p, \kappa \rangle$, composed of a proposal p, an argument set A_p and a function κ that classifies the arguments (as being in favour, neutral, or against the proposal).*

κ groups arguments in two separate sets: $A_p^+ = \{a \in A_p | \kappa(a) = 1\}$ includes those arguments in favour of the proposal and $A_p^- = \{a \in A_p | \kappa(a) = -1\}$ groups those against. Figure 2 b) depicts A_p^+ on the left column and A_p^- on the right. Neutral arguments (those with $\kappa(a) = 0$) are disregarded since they do not contribute much to the final decision.

An argument $a \in A_p$ is a pair $a = (s, O_a)$, where s is the argument description and O_a is the set of issued opinions. We then can combine these opinions into a single value rating the argument. We refer to this opinion aggregation as argument support ($S_{arg}(a)$), and compute it as a weighted mean. Weights in this function are computed by considering the opinions expressed using a 5-star scale, their number (see Figure 2 b)) and an importance opinion function meant to favour strong (clear) opinions over neutral positions —which can be associated with indecision (see [14] for further details).

This opinion support allows us to discard weak (non-relevant) arguments that would hinder the aggregated support. We consider an argument to be *relevant* if it has a significant number of opinions and it has enough support (significance is computed in terms of proportion of the issued opinions w.r.t. the maximum number of opinions issued for the other arguments in the debate).

Hence, PAM just focuses on relevant arguments in both A_p^+ and A_p^- and scales up the computation of the opinion support for the resulting sets $A_p^{R^+}$ and $A_p^{R^-}$. In this case, the aggregation operator we use is the WOWA (Weighted Ordered Weighted Average) operator [18] which, in addition to the importance of aggregated opinions over (relevant) arguments, it also considers their relative ordering (again, see [14] for a detailed explanation). We denote the support of these two argument sets as $S_{set}(A_p^{R^+})$ and $S_{set}(A_p^{R^-})$ respectively. However, it is worth noticing that we only compute the aggregated support of an argument set whenever there are relevant arguments. Otherwise, we consider we lack enough quality opinions to compute the aggregated opinion.

Finally, given a proposal p and the supports of the (non-empty) sets of relevant arguments in favour and against the proposal (i.e., $S_{set}(A_p^{R^+})$ and $S_{set}(A_p^{R^-})$ respectively), we compute the proposal support $S_{prop}(p)$ by applying the same WOWA operator (details can be found in [14]).

2.2. Target oriented discussion framework (TODF)

The Target Oriented Discussion Framework (TODF) [9] is an alternative support aggregation method to be used in debates. TODF focuses on the debate structure, which is specified in terms of arguments that can attack or defend both the proposal as well as other arguments.

Formally, a Target Oriented Discussion Framework is a structure $TODF = \langle \mathscr{A}, \mapsto, \vdash, \tau \rangle$, where $\mathscr{A}$ is a set of arguments; $\mapsto \subseteq \mathscr{A} \times \mathscr{A}$ is an attack relation (if $a \mapsto a'$, then a is attacking a'); $\vdash \subseteq \mathscr{A} \times \mathscr{A}$ is a defence relation (if $a \vdash a'$, then a is defending a') and τ is the target (in our case the proposal). Figure 2 c) shows how we represent attack and defence relations with red and green arrows respectively.

In our Target Oriented Discussion Framework, citizens express their opinions by assigning qualitative values (labels) to arguments and the proposal (which is a particular argument). Such labels are: `in` , meaning they agree with the argument; `out`, if they disagree; and `undec`, if they are neutral, not sure, or not defined. Each citizen i provides an argument labelling L_i, so we can define $\texttt{in}_L(\mathscr{A}) = |\{a \in \mathscr{A} \ |L_i(a) = \texttt{in}\}|$ as the number of arguments citizen i accepts (and analogously $\texttt{out}_L(\mathscr{A})$ for rejected arguments). Moreover, we define the collection of all individual opinions as a labelling profile $\mathscr{L}$, which contains all the individual argument labellings.

Subsequently, we compute the aggregated support of arguments –and in particular, of the proposal– by means of an aggregation function (AF) that exploits the argument relationships to combine and propagate argument opinions.

Firstly, given an argument $a \in \mathscr{A}$ and a labelling L, we consider its defending and attacking arguments as $D(a) = \{b \in \mathscr{A} | b \vdash a\}$ and $A(a) = \{c \in \mathscr{A} | c \mapsto a\}$ and define:

- Positive support of a by labelling L: $Pro_L(a) = in_L(D(a)) + out_L(A(a))$ adds the number of accepted defending arguments and rejected attacking arguments.
- Negative support of a by L: $Con_L(a) = in_L(A(a)) + out_L(D(a))$ represents the number of accepted attacking arguments plus rejected defending arguments.

Next, given an argument $a \in \mathscr{A}$ and a labelling profile $\mathscr{L}$, we compute both its Indirect Opinion (IO) and its Direct Opinion (DO) by considering the labels attached to the arguments a is related with:

$$IO(\mathscr{L})(a) = \begin{cases} 1 & \text{if } Pro_{AF(\mathscr{L})}(a) > Con_{AF(\mathscr{L})}(a) \\ 0 & \text{if } Pro_{AF(\mathscr{L})}(a) = Con_{AF(\mathscr{L})}(a) \\ -1 & \text{if } Pro_{AF(\mathscr{L})}(a) < Con_{AF(\mathscr{L})}(a) \end{cases} \quad DO(\mathscr{L})(a) = \begin{cases} 1 & \text{if } in_{\mathscr{L}}(a) > out_{\mathscr{L}}(a) \\ 0 & \text{if } in_{\mathscr{L}}(a) = out_{\mathscr{L}}(a) \\ -1 & \text{if } in_{\mathscr{L}}(a) < out_{\mathscr{L}}(a) \end{cases}$$

Finally, we asses the aggregated label of an argument a by applying the following aggregation function AF (see BF in [9]) which balances both direct and indirect support:

$$AF(\mathscr{L})(a) = \begin{cases} in & \text{if } IO(\mathscr{L})(a) + DO(\mathscr{L})(a) > 0 \\ out, & \text{if } IO(\mathscr{L})(a) + DO(\mathscr{L})(a) < 0 \\ undec, & \text{if } IO(\mathscr{L})(a) + DO(\mathscr{L})(a) = 0 \end{cases} \qquad (1)$$

3. Hybrid aggregation for Decidim

The two methods hereby presented tackle the aggregation of proposal support in alternative ways. On the one hand, PAM relies heavily on the quantitative (real-number) opinions that citizens express on arguments and filter out non-relevant arguments. On the other hand, TODF focuses on the dialogue structure (i.e., the attacking and defending relationships among arguments) and operates with qualitative (labelling) opinions. Therefore, we propose here a hybrid approach combining the strengths of both aggregation methods.

Inspired by the TODF aggregation function, we meant our hybrid aggregation method to grant the same importance to both direct and indirect opinions. However, Decidim just gathers positive supports for proposals to avoid possible harmful "hater" behaviours. Therefore, our hybrid aggregation function just considers indirect opinions over the proposal (i.e., those over the arguments issued when debating the proposal).

We resort to the PAM method to compute such indirect opinion so we benefit from a real-value spectrum and the possibility of discarding non-relevant arguments. However, PAM just considers arguments in favour and against the proposal, thus disregarding the tree structure of the debates in Decidim. In order to overcome this limitation, we first use the aggregation function AF in Eq. 1 from TODF to aggregate the opinions of every first-level subtree in the debate structure. These subtrees correspond to those trees whose roots correspond to the first-level arguments (i.e., arguments that are directly related – either by an attack or a defence relation– to the proposal). For instance, in Figure 2 c) there are 12 of such subtrees, although just 4 of them have more than a single argument.

Subsequently, the resulting aggregated qualitative opinions obtained for each first-level argument are transformed into real values by mapping `in` to 1, `out` to -1, and `undec` to 0. Finally, we are ready to use PAM to compute the final aggregated opinion on the proposal.

3.1. Hybrid aggregation results

We apply our hybrid method to real debates from Decidim Barcelona [3]. In previous work [16] we conducted a comparison of 910 proposals to assess the coincidences in support aggregation between PAM, TODF, and an base-line average computation. Results showed that they behave similarly, as about 95% match those of the average computation. Differences in the remaining c.a. 5% mostly come from: their ability to overcome average's tendency to centralise scores; PAM's aggregating quantitative opinions and deeming some proposals to be not evaluable due to lack of relevant information; and TODF relating qualitative opinions.

In order to provide some intuition of how our hybrid aggregation method[3] behaves, Table 1 explores 5 different proposals from [3]: the example proposal 50 from Figure 2 and 4 additional proposals. They have been chosen because they had high direct support (first row) and a significant number of arguments. Rows 2, 3, and 4 correspond to the number of arguments in favour, neutral, and against respectively that are first-level (in Decidim, all lower-level arguments are neutral and also added in row 5)[4]. We processed

[3] Source code by Marc Fernàndez is publicly available at https://github.com/marcFernandez/TODF-Argumentation/blob/tfg/NormArgumentMap.java.

[4] For instance, proposal 262 has a total of 111 arguments: 63 first-level and 52 (neutral) lower-level.

	proposal 50	prop. 179	prop. 262	prop. 1219	prop. 1256
direct support	57	446	705	141	1720
arg. in favour	11, **11**	51, **31**	31, **16**	10, **8**	49, **55**
neutral arg.	0, **0**	3, **0**	10, **0**	6, **0**	13, **0**
arg. against	1, **1**	0, **0**	22, **4**	14, **10**	8, **0**
total arg.	22, 12, **12**	55, 54, **31**	111, 63, **20**	68, 30, **18**	108, 70, **55**
Average	0.575	1.0	0.745	-0.380	0.945
PAM	0.715	0.995	0.885	-0.715	0.950
TODF	1 (in)	1 (in)	1 (in)	-1 (out)	1 (in)
Hybrid	0.75	0.98	0.9	-0.81	0.93

Table 1. Description of five different proposals from [3] and their corresponding aggregated support values as computed by Average, PAM, TODF, and Hybrid aggregation methods.

these arguments –to discard those without positive opinions or to realign those mistakenly defined– and show in bold the arguments actually used in the hybrid aggregation (see rows 2-5, where 5 adds previous rows 2-4).

Subsequent rows (from 6 to 9) in Table 1 provide the support computed by the different aggregation methods[5]. Overall, all methods coincide in the sign of the proposal aggregated support, although average shows its tendency to centrality. This is most noticeable in proposal number 1219, which is about "Limiting the invasion of dogs in public spaces". Average shows a much more bland negative value (-0.380) than PAM (-0.715), TODF (-1), or Hybrid (-0.81), which clearly result in rejecting the proposal. PAM discards most of the arguments and, since arguments against are much stronger, it rejects the proposal. As for TODF, it accepts (i.e, labels as in) more arguments against than in favour, so that it also rejects the proposal. This proposal 1219 also exemplifies well the tendency of our new hybrid method to result in values that are close to PAM and TODF values. In fact, this hybrid method results in a useful combination that inherits the advantages of each of the strenghts of PAM and TODF methods (i.e., PAM's real values allow to discard arguments that do not contribute much to the debate, and TODF's qualitative process exploits the debate structure).

4. Resilience

Debate quality has been studied based on different perspectives such as the structure of the debate (e.g., when measuring the depth of the thread of a debate [2]), the quantity of arguments issued, or number of gathered opinions. The debates we consider here are associated to decision making –that is, they are meant to decide upon the acceptance or rejection of a proposal– and thus, we propose *resilience* as a measure related to the quality (or robustness) of this decision. It is meant to perform a *sensitivity analysis* to determine how the decision is affected based on changes on the arguments and opinions. Specifically, the changes we consider are the addition of new counter-arguments and opposed opinions. In this manner, if a proposal is currently accepted, the resilience will measure the proportion of arguments (and opinions) against it that are required to be added so it becomes rejected (or conversely, if it is rejected, the changes required to

[5] For the sake of comparison, all values have been normalised to the interval [-1, 1]. Therefore, the ones for TODF can only take extreme values of -1, 0, or 1, whereas the rest of methods are more fine grained.

become accepted). Thus, the more changes needed, the more resilient the outcome of the debate (i.e., the decision made). In this manner, resilience can provide information to participants on how far the debate is (or what would it take to) reverse the current decision. Moreover, this would help participants that do not agree with the outcome to picture why their option is not chosen and to stimulate them to look for further participation (in terms of additional arguments or opinions).

In this context, we will refer to *aligned arguments* as those that are in line with the outcome of the debate. That is: if the proposal gets accepted, aligned arguments correspond to those in favour of the proposal. Otherwise, if the proposal is rejected, aligned arguments are those against it. Following TODF notation:

$$aligned\_arg(p) = \begin{cases} \{a \in \mathscr{A} | a \vdash p\} & \text{if } AF(p) = \texttt{in} \\ \{a \in \mathscr{A} | a \mapsto p\} & \text{if } AF(p) = \texttt{out} \end{cases}$$

Similarly, we will refer to *contrary arguments* are those that are opposite:

$$contrary\_arg(p) = \begin{cases} \{a \in \mathscr{A} | a \mapsto p\} & \text{if } AF(p) = \texttt{in} \\ \{a \in \mathscr{A} | a \vdash p\} & \text{if } AF(p) = \texttt{out} \end{cases}$$

Our resilience computes the sensitivity analysis by considering different types of argument and opinion additions. In particular, we define 4 debate extension strategies:

1. The addition of weak synthetic contrary arguments. By weak we mean that are minimally relevant (i.e., with enough opinions to be considered relevant as defined in Sect. 2.1). It measures the proportion of weak arguments that would be required to change the decision (i.e., how resilient it is w.r.t. new weak counter-arguments).
2. The addition of strong synthetic contrary arguments. By strong we mean having as many opinions as the aligned argument with the maximum number of opinions. Thus, this computes the sensitivity of the decision to strong arguments.
3. The addition of synthetic contrary arguments that are equivalent to, in terms of number of opinions, the best existing contrary argument. We only compute this dimension in case such contrary argument exists and it is relevant.
4. The addition of synthetic negative opinions about existing aligned arguments.

From these 4 debate extension strategies we define the so-called *resilience profile* as a vector $\vec{\rho}$ of four components, where each component reflects the proportion of additions that need to be included into a debate for it to change its outcome:

$$\vec{\rho}(p) = \langle \rho_1(p), \rho_2(p), \rho_3(p), \rho_4(p) \rangle$$

We compute resilience through an iterative process that consists on performing unitary synthetic additions to the debate and computing the resulting proposal support until this support changes (i.e., from accepted to rejected or vice-versa). Finally, we compute each resilience component $i \in [1,4]$ as the percentage of required synthetic additions (syn_add_i) with respect to the total elements ($total_i$) under consideration (i.e., total number of arguments for ρ_1, ρ_2, ρ_3 or total number of opinions for ρ_4):

$$\rho_i = min(100, \frac{syn\_add_i * 100}{total_i}) \tag{2}$$

4.1. Resilience results

This section is devoted to illustrate how our resilience measures the sensitivity of the decision made in a debate. Table 2 details the resilience profile for the proposals in Table 1. Furthermore, Figure 3 depicts the radar diagrams for proposals 50, 262, and 1219 (those for proposals 179 and 1256 are similar to 262 and 1219 respectively). Columns in Table 2 indicate the value (specified as a percentage) for each resilience dimension together with the values of syn_add_i and $total_i$ used in its computation. In particular, $total_1 \dots total_3$ correspond to the number of arguments specified in 5th row of Table 1, whereas $total_4$ specifies the total number of opinions issued for those arguments.

prop. p	$\rho_1(p)$ (syn_add_1,$total_1$)	$\rho_2(p)$	$\rho_3(p)$	$\rho_4(p)$(syn_add_4,$total_4$)
50	83,33% (10, 12)	91.67% (11, 12)	100% (0, 12)	100% (18 No-Comp, 30)
179	100% (33,31)	38.7% (12,31)	100% (0, 31)	100% (114 No-Comp, 184)
262	80% (16, 20)	55% (11, 20)	100% (0, 20)	100% (79 No-Comp, 120)
1219	33.3% (6,18)	16.7% (3,18)	100% (435,18)	100% (31 No-Comp, 75)
1256	27.3% (15,55)	20% (11,55)	100% (0, 55)	100% (38 No-Comp, 275)

Table 2. Resilience profiles for the five proposals from Table 1.

When applying Equation 2, different proposals result in different values of ρ_1, varying from the 100% for prop. 179 (because becomes rejected only when adding 33 out of 31 weak synthetic arguments against it) down to the 27.3% (15 out of 55) for prop. 1256. We can also observe that the percentages for ρ_2 tend to be smaller than ρ_1, since fewer contrary arguments are required to change the decision if they are strong[6] (recall that arguments added in ρ_1 are the weakest possible). As for ρ_3, most contrary arguments turn out to be irrelevant and, thus, ρ_3 becomes the maximum possible (i.e., 100%)[7]. Finally, as for ρ_4, it turns out to be the case that for all proposals in Table 2, the iterative process goes as far as adding a number of negative opinions about aligned arguments that causes the proposal support to become non-computable (see Subsection 2.1). Again, we also interpret this situation as a maximum resilience and assign $\rho_4(p) = 100\%$.

Overall, we argue that these proposal examples (and in particular, prop. 1219 and 1256) illustrate well the need for the resilience measure, since a debate outcome that may seem at first sight quite clear (recall, from last 3 rows in Table 1, that our support aggregation methods computed values above 0.7 in absolute value) could in some cases be in fact prone to change if relatively few new contrary arguments were added.

5. Conclusions and Future Work

This paper proposes a new support aggregation method meant to be used by the Decidim participation platform. This method is derived from the combination PAM and TODF, two complementary aggregation methods previously introduced in the literature. This hybrid method inherits: from PAM, the fine-grained real values that allow to discard irrelevant arguments in the debate; and from TODF's, the qualitative process that exploits

[6] E.g., in 179 the strongest argument has 15 positive opinions and none negative ones.

[7] The exception being prop. 1219, where we need to add as much as 435 synthetic arguments because the best contrary argument is rather controversial (it has 4 opinions in favour and 3 against).

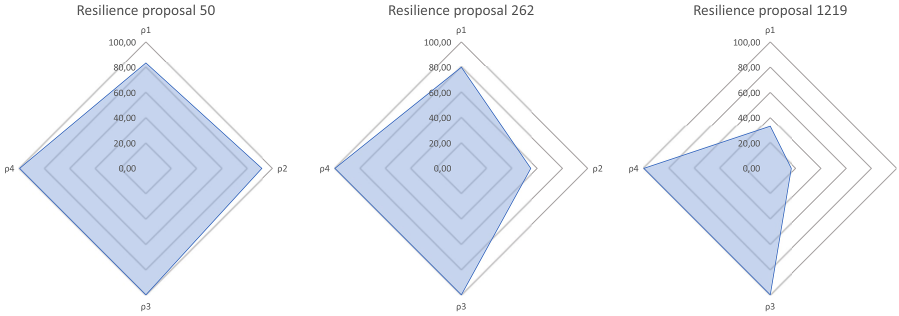


Figure 3. Radar diagrams of the resilience profiles for proposals 50, 262, and 1219.

the debate structure. Additionally, we propose a resilience metric for measuring the quality of the aggregated opinion and illustrate its application to debates conducted in the Decidim participatory platform. As future work, we plan to integrate both the hybrid aggregation method and the resilience measures in the Decidim open source platform. The literature on consensus in aggregation functions also deserves future exploration.

References

[1] City of Reykjavík participation portal. http://reykjavik.is/en/participation, 2021. last visited.

[2] Pablo Aragón, Vicenç Gómez, and Andreas Kaltenbrunner. To thread or not to thread: The impact of conversation threading on online discussion. In *11th Int. Conference on Web and Social Media*, 2017.

[3] Decidim Barcelona. Argument data on debates in the municipal action plan participation process. https://github.com/elaragon/metadecidim/tree/master/comments, 2016.

[4] Decidim Barcelona. City of barcelona participation portal. https://decidim.barcelona, 2021. last visited.

[5] Consul. Free software for citizen participation. http://consulproject.org/en/, 2021. last visited.

[6] Decidim. Participatory democracy for cities and organizations. https://decidim.org/, 2021. last visited.

[7] Charles DeTar. *InterTwinkles: Online Tools for Non-Hierarchical Consensus-Oriented Decision Making*. PhD thesis, Media Arts and Sciences at the Massachusetts Institute of Technology, 2013.

[8] Parlement et Citoyens. Institutional participation portal for the discussion of french laws. https://parlement-et-citoyens.fr/, 2019.

[9] Jordi Ganzer-Ripoll, Natalia Criado, Maite Lopez-Sanchez, Simon Parsons, and Juan A. Rodriguez-Aguilar. Combining social choice theory and argumentation: Enabling collective decision making. *Group Decision and Negotiation*, 28(1):127–173, Feb 2019.

[10] Spaceship Media. Spaceship media: journalism to bridge divides. https://spaceshipmedia.org, 2016. last visited: May 2021.

[11] Participa Gencat portal. https://participa.gencat.cat/, 2021. last visited.

[12] Carole Pateman. *Rousseau, John Stuart Mill and G. D. H. Cole: a participatory theory of democracy*, page 22–44. Cambridge University Press, 1970.

[13] Petitions. UK Government and Parliament. https://petition.parliament.uk/, 2021. last visited.

[14] Juan A Rodriguez-Aguilar, Marc Serramia, and Maite Lopez-Sanchez. Aggregation operators to support collective reasoning. In *Modeling Decisions for Artificial Intelligence*, pages 3–14. Springer, 2016.

[15] Rousseau platform. https://rousseau.movimento5stelle.it/, 2021. last visited.

[16] Marc Serramia, Jordi Ganzer, Maite López-Sánchez, Juan A. Rodríguez-Aguilar, Natalia Criado, Simon Parsons, Patricio Escobar, and Marc Fernández. *Citizen Support Aggregation Methods for Participatory Platforms*, volume 319, page 9–18. IOS Press, 2019.

[17] Frank J Sorauf. The public interest reconsidered. *The Journal of Politics*, 19(4):616–639, 1957.

[18] Vicenç Torra and Yasuo Narukawa. *Modeling decisions: information fusion and aggregation operators*. Springer Science & Business Media, 2007.

[19] Vishanth Weerakkody and Christopher G Reddick. *Public sector transformation through e-government: experiences from Europe and North America*. Routledge, 2012.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210164

Contextualized LORE for Fuzzy Attributes

Najlaa MAAROOF [a,1], Antonio MORENO [a], Mohammed JABREEL [b] and Aida VALLS [a]

[a] *ITAKA-Intelligent Technologies for Advanced Knowledge Acquisition - Departament d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili, Tarragona, Spain*

[b] *Microsoft Advanced Technology Lab, Cairo, Egypt*

Abstract. Despite the broad adoption of Machine Learning models in many domains, they remain mostly black boxes. There is a pressing need to ensure Machine Learning models that are interpretable, so that designers and users can understand the reasons behind their predictions. In this work, we propose a new method called *C-LORE-F* to explain the decisions of fuzzy-based black box models. This new method uses some contextual information about the attributes as well as the knowledge of the fuzzy sets associated to the linguistic labels of the fuzzy attributes to provide actionable explanations. The experimental results on three datasets reveal the effectiveness of *C-LORE-F* when compared with the most relevant related works.

Keywords. Machine Learning, Fuzzy Logic, Explanation AI

1. Introduction

The past decade has witnessed significant improvements in the development of Machine Learning (ML)-based systems, due to the presence of huge amounts of data, the advances in Deep Learning and the affordability of more advanced computer equipment. As a result, ML has become a vital component of multiple applications in many fields. However, most ML-based systems are considered black boxes, and it is not straightforward to understand the reasons behind their decisions. Hence, developing methods for interpreting the decisions of such systems has become highly demanded [1, 2].

Explainability mechanisms for complex ML models can be *model-dependent* or *model-agnostic*. The former analyse the internal structure of the model (e.g. the weights inside a neural network) to understand their internal working and come up with a clear picture of how the solution is computed. The latter usually generate a set of inputs, analyse the answers provided by the black box to them, and then create a simpler model from which we can infer an explanation [3–5].

[1] Corresponding Author: Najlaa Maaroof. E-mail: najlaamaaroofwahib.al-ziyadi@urv.cat.

Two well-known post-hoc explanation methods are *Local Interpretable Model-agnostic Explanations* (LIME, [5]) and *Local Rule-Based Explanations* (LORE, [6]). LIME samples points randomly from the same distribution of the point to be explained, sends them to the black box, and provides local explanations for the classifier's prediction by fitting a linear regression model locally around that point. LORE generates a set of neighbours of the input point using a genetic algorithm, applies the black-box on each of them, and builds a decision tree on these results. The LORE explanation contains the rule used to produce the decision and a set of *counterfactual rules* which represent the minimal number of changes in the feature values of the instance that would change the conclusion of the system.

Despite the popularity of LORE and LIME, they have some shortcomings. First, they do not use any knowledge about the characteristics of the input features. Secondly, the neighbour generation algorithms generate"close" points more or less blindly. In our previous work [7] we proposed *Guided-LORE* (or *G-LORE*), a version of LORE in which the neighbourhood generation is formalised as a search problem and solved using Uniform Cost Search, making it more focused. However, like LIME and LORE, Guided-LORE does not consider explicitly the case in which the attributes that define the objects are fuzzy. For example, when we use Guided-LORE, a neighbour of a point is generated by adding (or substracting) a fixed amount (which is called *step*) to the value of an attribute. In the case of using fuzzy attributes this option could be reasonable if the fuzzy sets associated with the linguistic labels were equally and uniformly distributed across all the domain. However, in many situations, this property does not hold.

This paper proposes *C-LORE-F* (Contextualized LORE for Fuzzy attributes), a variant of Guided-LORE that addresses those issues. Our first motivation is that, if we know that an attribute is fuzzy and we have the information on its fuzzy labels and their associated fuzzy sets, we can profit from that knowledge to make a more focused neighbourhood generation. More precisely, we can generalise its step from being a fixed value to being a function that depends on that knowledge. In that way the proposed method is more general, and it works in the cases in which the fuzzy sets associated with the linguistic labels are uniformly or non-uniformly distributed. To the best of our knowledge, this work is the first one that utilises such knowledge to develop explanation methods for ML systems based on fuzzy logic. The second novel point of the system is the use of the knowledge about the type of attribute to guide the neighbourhood generation process and search for *actionable* explanations. For example, if we have an attribute like age, which automatically increases in time, it probably does not make much sense to look for neighbours that have a lower value in this attribute, as it would not be very interesting for the user to receive an explanation like "if you were 10 years younger, the prediction of the system would be different" (even if this explanation was correct). This knowledge about the type of attribute is not currently being used in the most popular explanation systems.

The rest of this article is structured as follows. Section 2 provides an overview of the related works. Section 3 explains the proposed neighbourhood generation procedure and the types of attributes that have been considered. In Section 4 we describe the experimental setup and discuss the obtained results. Finally, in section 5, we conclude the paper and list some points for future work.

2. Related Work

In the last years the research of methods for explaining black box decision systems has received a lot of attention [8], and there has been an extraordinary amount of articles in ML interpretability in the last years. This section comments some of the most relevant ones to this work. We can categorise the works on ML interpretability into those based on features' importance (§2.1), counterfactual examples (§2.2) and visualisation mechanisms (§2.3).

2.1. Feature relevance

There are two main directions for developing explanation methods based on the importance of features, *global* and *local* explanation methods. Global methods try to explain the entire model behaviour using surrogate models whereas local models explain a single prediction. It can be useful, in some scenarios, to understand the global logic of a model. However, the major issue with such approaches is that, as the explanations are extracted from simpler surrogate models, there is no guarantee that they are faithful to the original model [3, 5, 9].

Local explanation methods are considered to be one of the fundamental approaches to post-hoc explanations. *LIME* [5], already mentioned in the introduction, is a well-known example of such approaches. It is independent of the type of data and the black box to be explained. Given a black box model f, an instance x, and a decision y produced by f on x, LIME constructs a simple linear model that approximates f's input-output behaviour to justify why f predicts y. It generates some neighbours of x randomly in the feature space, that are centred on x. Such an approach is becoming a conventional method. We can find now LIME implementations in multiple popular packages, including Python, R and SAS.

The authors of LIME observed that it does not measure its fidelity. As a result, the local behaviour of a notably non-linear model may lead to faulty linear approximations. Hence, they were motivated to work on a new model-agnostic method, *Anchors*, based on if-then rules [10]. This method highlights the part of the input that is adequate for the classifier to make the prediction, delivering more intuitive and easy-to-understand explanations.

SHAP [11] is a method that provides an explanation of the prediction of the output of a black box for an instance x by estimating each feature's contribution to that prediction. These contributions are collected by measuring the *Shapley values* from coalitional game theory. The features act like players in a coalition. Each player can be formed by a single feature or a subset of features. The Shapley values show the payout distribution of the prediction among the features.

2.2. Counterfactual examples

Another important category of explanations is based on the generation of counterfactuals. These methods seek minimal changes to the feature values such that the model's predicted outcome changes. These changes should actually be actionable to be useful (e.g. an applicant for a bank loan might want to know which part of her application could be changed to get her application approved).

The pioneering work by Martens and Provost [12] used a best-first search to develop a model-agnostic method for finding counterfactuals that explain the predictions of any classifier. LORE [6] is another example of this approach, although it can also be seen as an example of the determination of features' importance. It constructs a decision tree c based on a synthetic neighbourhood of the input point. Then, an explanation e, composed of a decision rule and a set of counterfactual rules, based on some extracted counterfactual examples, is obtained from the logic of c. The work presented in [13] proposed a general optimisation framework to generate sets of diverse counterfactual examples for any differentiable ML classifier. Russell [14] proposed in 2019 a "mixed polytope", a set of constraints that can be used with integer programming solvers to extract counterfactual explanations without making a brute-force enumeration.

2.3. Methods focused on visualisation

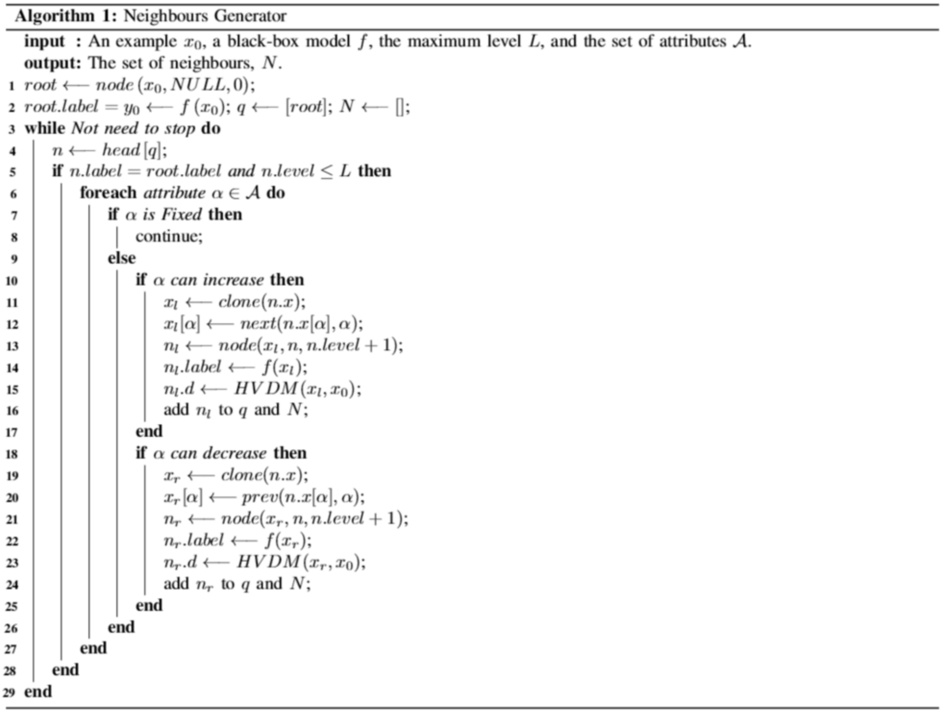
We can find several visualisation methods proposed in the literature to help ML engineers and domain experts to understand, debug, and refine ML models. For example, Ming et al. [15] proposed an interactive visualisation method to help users, even those without expertise in Machine Learning, to understand, explore and confirm predictive models. This method by Ming et al. extracts a set of rules that approximates a classifier's prediction and visualises them using an interactive visual interface. Neto and Paulovich proposed Explainable Matrix (ExMatrix) [16], a visualisation method to interpret Random Forests. They used a matrix-like visual metaphor in which rows represent rules, columns are features, and cells are rules predicates. They showed that their method is capable of offering global and local explanations of Random Forest models.

Like LORE and Guided-LORE, our proposal can be considered as a hybrid of the approaches based on features' importance and counterfactual examples. However, unlike them, it focuses on the case in which the attributes that define the objects are fuzzy.

3. Contextualised LORE for Fuzzy attributes

C-LORE-F uses contextual information (the type of attribute and the fuzzy sets associated to the linguistic values of the fuzzy attributes) to produce explanations. Its inputs are a trained fuzzy-based ML model, f, and an example x_0. First, we apply f to x_0 to get a decision y_0. Then, we apply *C-LORE-F* to generate an explanation, which is composed of a rule r and a set of counterfactual rules δ that produce a different outcome. To this end, the general LORE process is used:

1. Generate a set of neighbours $\mathcal{G}$ of x_0.
2. Train a decision tree t using $\mathcal{G}$.
3. Inspect t and extract the rule r used to classify x_0,
4. Generate a set of counterfactual examples to x_0, pass them to t to get their labels and get the set of counterfactual rules δ.



```
Algorithm 1: Neighbours Generator
input : An example $x_0$, a black-box model $f$, the maximum level $L$, and the set of attributes $\mathcal{A}$.
output: The set of neighbours, $N$.
$root \longleftarrow node\,(x_0, NULL, 0)$;
$root.label = y_0 \longleftarrow f\,(x_0)$; $q \longleftarrow [root]$; $N \longleftarrow []$;
while Not need to stop do
    $n \longleftarrow head\,[q]$;
    if $n.label = root.label$ and $n.level \leq L$ then
        foreach attribute $\alpha \in \mathcal{A}$ do
            if $\alpha$ is Fixed then
                continue;
            else
                if $\alpha$ can increase then
                    $x_l \longleftarrow clone(n.x)$;
                    $x_l[\alpha] \longleftarrow next(n.x[\alpha], \alpha)$;
                    $n_l \longleftarrow node(x_l, n, n.level + 1)$;
                    $n_l.label \longleftarrow f(x_l)$;
                    $n_l.d \longleftarrow HVDM(x_l, x_0)$;
                    add $n_l$ to $q$ and $N$;
                end
                if $\alpha$ can decrease then
                    $x_r \longleftarrow clone(n.x)$;
                    $x_r[\alpha] \longleftarrow prev(n.x[\alpha], \alpha)$;
                    $n_r \longleftarrow node(x_r, n, n.level + 1)$;
                    $n_r.label \longleftarrow f(x_r)$;
                    $n_r.d \longleftarrow HVDM(x_r, x_0)$;
                    add $n_r$ to $q$ and $N$;
                end
            end
        end
    end
end
```

Figure 1. Neighbours Generator.

For more details about the counterfactual examples generation and rules extraction, we refer the reader to [6]. The set $\mathcal{G}$ is obtained by merging two subsets, $\mathcal{G}^+$ and $\mathcal{G}^-$. The first one is called the *positive set*, and it contains a set of instances that belong to the same class of x_0. We get this subset by passing the instance x_0 to Algorithm 1. The second one, the *negative set*, contains examples with a different class. We obtain $\mathcal{G}^-$ by looking at an auxiliary set T and finding the closest example to x_0 , i.e., x_0^-, that has a different label than y_0. T can be the training set used to train the black-box model, if accessible, or any other data set from the same distribution. Once we get x_i^-, we pass it to Algorithm 1 to generate the *negative set*.

The *Neighbours Generation* step aims to find the set $\hat{\mathcal{G}}$ with the points that are close to a given instance $\hat{x}$ and have the same class. $\hat{\mathcal{G}}$ can be either $\mathcal{G}^+$ if $\hat{x} = x_i$ or $\mathcal{G}^-$ if $\hat{x} = x_i^-$.

As a first change with respect to LORE and Guided-LORE, we have defined the following types of attributes.

- Attributes with a fixed value (e.g. sex).
- Attributes whose value increases in time (e.g. age).
- Attributes whose value decreases in time (e.g. years left until retirement).
- Variable attributes, that can change positively and negatively (e.g. weight).

Neighbourhood generation is defined as a search problem in which we explore the neighbourhood space of a point $x_0 = \hat{x}$ by applying a Uniform Cost Search based on the Heterogeneous Value Difference Metric (HVDM, [17]), using some contextual information about the attributes (the attribute range and the step value). This search problem can be formulated as follows:

- **State Space**: the set of all possible examples S.
- **Initial State**: (x_0, y_0), where x_0 is the instance of which we want to generate its neighbours and y_0 is the label of this instance obtained by the black-box f.
- **Actions**: Modifications of the value of a single attribute (feature). These actions leverage some contextual information about the feature to make the desired changes to generate new neighbours. In our case we define two types of actions, *next* and *prev*, described later.
- **Transition Model**: returns a new instance in which the value of a feature is changed by applying all actions.
- **Goal Test**: We check, for each generated individual, if, according to the black box, it has the same label as the root, y_0. If that is the case, we generate its neighbours in the same way (i.e. applying one positive/negative change in the value of a single attribute). Otherwise, we have found an individual close to x_0 that belongs to another class; thus, we have reached a boundary of y_0, and we terminate the search from that instance.
- **Path Cost**: The path cost of each example is calculated by measuring the HVDM distance between the generated example and x_0.

Algorithm 1 shows how the neighbours of a given instance, x_0, are generated. The search tree starts in x_0, and in each node all the possible actions to move from one instance to another are applied. For each feature f, the number of possible actions can be zero (f is *Fixed*), one (either *next* if the feature is temporally increasing or *prev* if it is temporally decreasing) or both, if f is variable. Each action only changes the value of one feature. The candidate node to be expanded, n, is the one closest to x_0, based on the path cost. If the outcome of the black-box model for n is different from y_0, then it is a leaf of the tree. Otherwise, we expand that node. Consequently, for each node in the second level, we would have changes in two attributes or double changes in the same attribute, and so on. The expanding process finishes when we reach a predefined max-level, or when there are no more nodes to be expanded (all the leaves have led to changes in the initial classification). Repeated instances are ignored to avoid cycles.

The expanding process is done by cloning the instance of the current node, i.e., $n.x$ (lines 11 and 19 in Algorithm 1) and applying the *next* and/or *prev* actions. After that, we pass the obtained instance to the black box model f to get its corresponding label. To apply the actions *step* and *prev* for a given attribute we consider some *separate zones* based on its fuzzy sets, which are defined as shown in Figure 2, taking into account the intersection point between two consecutive fuzzy sets and the intervals of maximum activation. In Figure 2 the zones would be 0-5, 5-10, 10-15, 15-20, 20-25, 25-40, 40-50, 50-60, 60-75, 75-90 and 90-100. Given the value of the attribute, we locate its zone, and then we take the middle of the previous zone as the lower neighbour (the result of the *prev* action), and the middle of the next zone as the upper neighbour (the result of the *next* action). Figure 2 shows an example. The input value is 22, which belongs to the zone 20-25. Thus, the middle of the previous zone is the lower neighbour, (15 + 20)/2 = 17.5, and the middle of the next zone is the upper neighbour, (25 + 40)/2 = 32.5. We might end up applying only either the *next* action, if the located zone was the first one, or the *prev* action, if it was the last one.

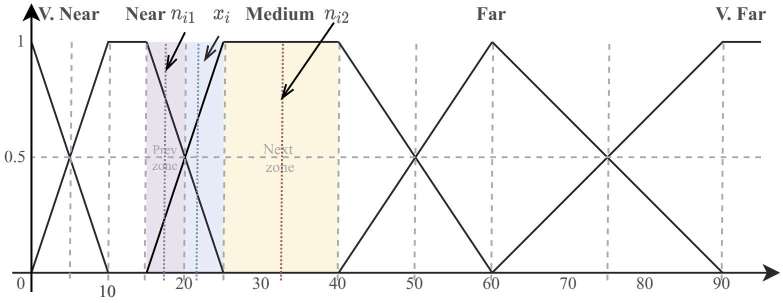


Figure 2. Illustration of the *next* and *prev* actions.

4. Experiments and Results

4.1. Experimental Setup

We used three data sets in our experiments: German-Credit, Adult-Income and Diabetic-Retinopathy. The first two ones are publicly available in the well-known UCI Machine Learning Repository, whilst the last one is a private data set for the assessment of the risk of developing diabetic retinopathy (DR) for diabetic patients. All of them are examples of binary classification. Considering the public data sets, as in [6], each data set was randomly split into a training set with 80% instances, and a test set, i.e., the set of instances for which the black box decision has to be explained, with 20% instances. The black box predictors used in the test were Fuzzy Random Forest (FRF) and Fuzzy Decision Tree (FDT). In case of Diabetic-Retinopathy, we directly used our fuzzy random forest-based system, Retiprogram, that is currently being used in the Hospital de Sant Joan in Reus (Tarragona). Table 1 illustrates the number of features and the number of training and testing examples used in the test for each data set. It also shows the accuracy scores of the black-box models for each data set.

Table 1. Data sets used in the experiments.

	Features	Train	Test	Total	Acc. FRF	Acc. FDT
Adult-Income	8	39,074	9768	48,842	0.781	0.654
German-Credit	20	8,000	2,000	10,000	0.715	0.715
Diabetic-Retinopathy	9	1,212	1,111	2,323	0.804	0.781

4.2. Evaluation Metrics

We used the following evaluation metrics:

- **hit**: this metric computes the accuracy between the output of the decision tree t and the black box model f for the testing instances. It returns 1 if $t(x)$ is equal to $f(x)$ and 0 otherwise.
- **fidelity**: this metric measures to which extent the decision tree accurately reproduces the black-box predictor, by comparing its predictions and the ones of the black-box on the instances $\mathcal{G}$.
- **l-fidelity**: it is similar to the fidelity, but it is computed on the instances covered by a decision rule in a local explanation for x. It is used to measure to what extent the rule is good at mimicking the black-box model.
- **c-hit**: it compares the predictions of the decision tree and the black-box model on a counterfactual instance of x that is extracted from the counterfactual rules in a local explanation of x.
- **cl-fidelity**: it is also similar to the fidelity, but it is computed on the instances covered by the counterfactual rules in a local explanation for x.

4.3. Results and Discussion

Table 2 shows the means and standard deviations of the metrics for C-LORE-F, LORE and G-LORE on the three data sets with the FRF and FDT models. In general, C-LORE-F outperforms the other methods in all metrics with the FRF model. In the case of FDT, it shows better performance than LORE and G-LORE in *hit* and *fidelity*. In the other metrics it is very close to the best one. LORE is the worst in most cases, especially with the FRF model. The reason is that G-LORE and C-LORE-F try to find the closest "frontier" between the class of x_0 and the other classes, producing a clearer decision boundary.

Table 2. The results on the three datasets.

Model	Method	*hit*	*fidelity*	*l-fidelity*	*cl-fidelity*	*c-hit*
FRF	LORE	0.96±0.19	0.98±0.02	0.97±0.07	0.45±0.43	0.43±0.39
	G-LORE	0.99±0.02	0.99±0.02	0.99±0.03	0.52±0.43	0.47±0.40
	C-LORE-F	1.00±0.0	0.99±0.0	0.99±0.0	0.59±0.39	0.58±0.42
FDT	LORE	0.95±0.22	0.98±0.03	0.98±0.03	0.48±0.41	0.45±0.44
	G-LORE	0.98±0.10	0.98±0.01	0.85±0.24	0.54±0.45	0.50±0.48
	C-LORE-F	0.99±0.05	0.99±0.0	0.97±0.08	0.43±0.43	0.41±0.46

Focusing on the black-box dimensions, all the methods show a better performance with the FRF in the *hit*, *fidelity* and *l-fidelity* metrics. This can lead us to conclude that the accuracy of a model is crucial in getting a better explanation.

At the data sets level, as shown in Figure 3, the best performance is obtained by Diabetic-Retinopathy, followed by Adult-Income. The reason is that all the explanation methods are sensitive to the accuracy of the black-box model. The more accurate is the model, the best is the obtained explanation, as confirmed by the *accuracy* scores reported in Table 1. In terms of c-hit and cl-fidelity, the best results are obtained with the Diabetic-Retinopathy data set. We can attribute

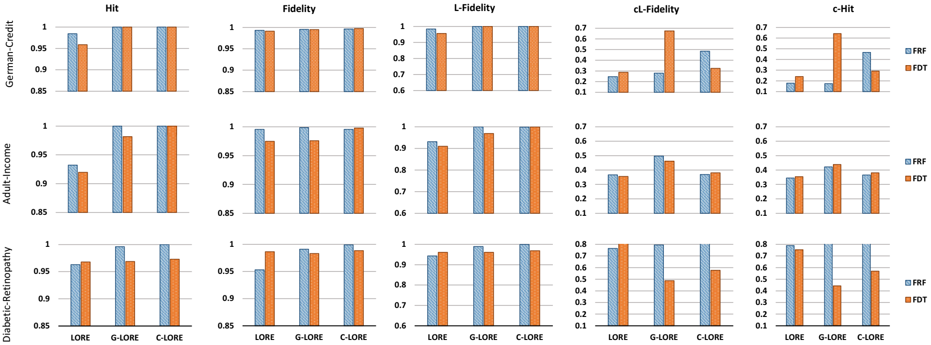


Figure 3. Comparison results: **LORE** vs **G-LORE** vs **C-LORE-F**.

this fact to the quality design of the fuzzy sets in this problem. The fuzzy sets of the Diabetic-Retinopathy data set were defined by an expert of the domain, whereas the fuzzy sets of Adult-Income and German-Credit were obtained automatically by applying a fuzzification algorithm [18]. The argument here is that these two metrics rely on the quality of the counterfactual examples that are used to generate counterfactual rules (which may be affected by the generated neighbours). Moreover, the intelligent design of the fuzzy sets is also a key factor in C-LORE-F as it utilises them as contextual information in the neighbourhood generation process. This can be confirmed by comparing the results of C-LORE-F vs others on the Diabetic-Retinopathy data set and comparing the performance of *C-LORE-F* method on the Diabetic-Retinopathy data set vs the other data sets. C-LORE-F outperforms LORE and G-LORE in almost all evaluation metrics. The cl-fidelity and c-hit are exceptions with the FDT and German-Credit case. In general, all the explanation methods showed a poor performance in terms of cl-fidelity and c-hit. That may be due to the bad quality of the counterfactual examples, and we intend to analyse this issue in our future work.

5. Conclusion

This paper has presented C-LORE-F, a new method to explain the decisions of fuzzy-based systems, that uses the information about the fuzzy sets that define the meaning of the linguistic values of the fuzzy attributes. It also considers the character of the attribute (whether its value is fixed, increasing, decreasing or variable). Its main advantage is that the generation of neighbours for a point x is more informed due to the usage of contextual information. Moreover, we search for boundaries with relevant meanings for the user (e.g. in order to avoid creating counterfactuals that depend on the change of a fixed attribute, or on the positive change of an attribute that only decreases on time). The experimental results on different data sets demonstrate the effectiveness of the proposed method. It outperformed the state-of-the-art methods in several metrics. The main issue in the proposed method is its poor performance on the c-hit and cl-fidelity metrics, although it showed a performance comparable to the best one. In our future work, we will focus on resolving this issue by improving the counterfactual examples generation. We should study the relationships between attributes to avoid generating impossible instances. Moreover, we intend to study how we can extract the explanations directly from fuzzy decision trees and use them as surrogate models.

Acknowledgements

This work has been funded by the research project PI18/00169 from Instituto de Salud Carlos III & FEDER funds and URV grant 2019PFR-B2-61. The first author is supported by a URV Martí Franquès predoctoral grant.

References

[1] Adadi A, Berrada M. Peeking inside the black-box: a survey on explainable artificial intelligence (XAI). IEEE access. 2018;6:52138–52160.

[2] Carvalho DV, Pereira EM, Cardoso JS. Machine learning interpretability: A survey on methods and metrics. Electronics. 2019;8(8):832.

[3] Burkart N, Huber MF. A Survey on the Explainability of Supervised Machine Learning. Journal of Artificial Intelligence Research. 2021;70:245–317.

[4] Krause J, Perer A, Ng K. Interacting with predictions: Visual inspection of black-box machine learning models. In: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems; 2016. p. 5686–5697.

[5] Ribeiro MT, Singh S, Guestrin C. 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. ACM; 2016. p. 1135–1144.

[6] Guidotti R, Monreale A, Ruggieri S, Pedreschi D, Turini F, Giannotti F. Local rule-based explanations of black box decision systems. arXiv preprint arXiv:180510820. 2018;.

[7] Maaroof N, Moreno A, Valls A, Jabreel M. Guided-LORE: Improving LORE with a Focused Search of Neighbours. In: Heintz F, Milano M, O'Sullivan B, editors. Trustworthy AI - Integrating Learning, Optimization and Reasoning. Springer; 2021. p. 49–62.

[8] Guidotti R, Monreale A, Ruggieri S, Turini F, Giannotti F, Pedreschi D. A survey of methods for explaining black box models. ACM computing surveys. 2018;51(5):1–42.

[9] Molnar C. Interpretable Machine Learning. Lulu.com; 2020.

[10] Ribeiro MT, Singh S, Guestrin C. Anchors: High-Precision Model-Agnostic Explanations. In: AAAI. vol. 18; 2018. p. 1527–1535.

[11] Lundberg SM, Lee SI. A unified approach to interpreting model predictions. In: Advances in neural information processing systems; 2017. p. 4765–4774.

[12] Martens D, Provost F. Explaining data-driven document classifications. Mis Quarterly. 2014;38(1):73–100.

[13] Mothilal RK, Sharma A, Tan C. Explaining machine learning classifiers through diverse counterfactual explanations. In: Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency; 2020. p. 607–617.

[14] Russell C. Efficient search for diverse coherent explanations. In: Proceedings of the Conference on Fairness, Accountability, and Transparency; 2019. p. 20–28.

[15] Ming Y, Qu H, Bertini E. Rulematrix: Visualizing and understanding classifiers with rules. IEEE transactions on visualization and computer graphics. 2018;25(1):342–352.

[16] Neto MP, Paulovich FV. Explainable MatrixVisualization for Global and Local Interpretability of Random Forest Classification Ensembles. IEEE Transactions on Visualization and Computer Graphics. 2020;.

[17] Wilson DR, Martinez TR. Improved heterogeneous distance functions. Journal of artificial intelligence research. 1997;6:1–34.

[18] Yuan Y, Shaw MJ. Induction of fuzzy decision trees. Fuzzy Sets and systems. 1995;69(2):125–139.

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210165

On the Rationality of Explanations in Classification Algorithms

Zoe FALOMIR [a,b] Vicent COSTA [c,d,1],
[a] *Universitat Jaume I, ES Tecnologia i Cincies Experimentals*
[b] *Bremen Spatial Cognition Center (BSCC)*
[c] *Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS), Universidade de Santiago de Compostela*
[d] *Artificial Intelligence Research Institute (IIIA-CSIC)*

Abstract. This paper is a first step towards studying the rationality of explanations produced by up-to-date AI systems. Based on the thesis that designing rational explanations for accomplishing trustworthy AI is fundamental for ethics in AI, we study the rationality criteria that explanations in classification algorithms have to meet. In this way, we identify, define, and exemplify characteristic criteria of rational explanations in classification algorithms.

Keywords. Rationality, rational explanation, explainable AI, classification algorithm, AI ethics, trustworthy AI

1. Introduction

Explainability is a fundamental topic for AI ethics. It increases users' trust in the outcomes of AI systems, eases to clarify the decisions made and whether the system has been trained on a biased/fair view of the world. But to accomplish these ethical goals, explanations need to be rational. In this way, this paper is a preliminary study of the criteria that rational explanations produced by classification algorithms must meet. Furthermore, examples of explanations produced by AI systems are extracted from research works in the state-of-the-art and used for discussion. We envision that these criteria may enable us to give a measure of how rational the explanations produced by AI systems are.

1.1. Trustworthy AI needs rational explainability

Broadly speaking, explainable artificial intelligence (XAI) is the discipline that studies those systems that describe their results, actions, or decisions. Regarding explainable classification algorithms, XAI yields reasons for the classification results obtained. Note that the lack of explainability decreases the trust in the outcomes of the AI systems, it reduces its fairness (without explainability, responsibility cannot be claimed) and usability [1], and makes it more probable to overlook whether they have been trained using a biased view of the world [2,3] (for a reference on biases in AI see [4,5]).

[1] Corresponding Author: vicent.costa@protonmail.com.

In the literature, AI systems that describe their outcomes in natural language are numerous and involve very diverse domains as finance, medicine, or social services. In this way, the demand for accountability regarding undesirable outcomes produced by algorithms requires the use of XAI; but not only that, explainability involves several topics as safety, transparency, or fairness. In addition, the General Data Protection Regulation[2], established by the European Parliament, includes the individual right to explanation, which affects AI systems and their users. Considering all these reasons, it is clear that XAI becomes a mandatory project.

However, not all descriptions of the outcomes appearing in the literature can be considered as adequate explanations. Similarly to human-human interactions, simple/incomplete/irrational descriptions might not be enough to discover bias in the design or training of the AI system, so they will not help to enlighten unethical results of their classification systems. Also, we cannot require trust if users do not find rationality in the explanations provided by algorithms. Therefore, providing rational explanations is a desirable property for AI classification models, and the present paper is a preliminary work on that direction.

1.2. How much does the ball cost? Biased thinking vs. rational thinking

In the literature, there are research studies that distinguish between two types of cognitive processes: those executed quickly with little conscious deliberation and those that are slower and more reflective [6,7,8,9]. These type of processes were called *System 1* and *System 2*, respectively [10]. *System 1* processes occur spontaneously and not require or consume much attention, i.e., recognizing a face, whereas *System 2* processes involve mental operations that require effort, concentration, and probably the execution of some learned rules, i.e., calculating 123x45 without a calculator. Note that the task 123x45 offers no intuitive solution, that is, no number spontaneously comes to your mind as a possible answer. Let us exemplify these two type of processes with an easy question:

A bat and a ball cost 1.10 €. The bat costs 1.00 € more than the ball.

How much does the ball cost? ___ cents.

This is a question in the Cognitive Reflection Test (CRT) [11]. In this case, individuals who answer this question with the first idea that comes to mind usually fail this test[3]. They are also biased reasoners since they are quite convinced that the problem is very easy since they estimate that 92% of the people would solve this problem right [11]. This exemplifies that we people can be mistaken and unaware, specially in situations where we think superficially. That is a flaw in the reflective mind, a failure of our rationality.

Moreover, as [9] mentions, people are supposed to have consistent preferences that they trust since they reflect their interests. But sometimes our decisions do not produce the best possible experience for us. Note that taste and decisions are shaped by memories and the duration of these memories can be neglected/biased. For example, usually people give the same importance/weight to the good and to the bad part of an experience, although the good part have lasted ten times longer than the other.

Thus, AI systems that produce rational explanations may also help individuals to realise other sides of their judgment that might be biased, so that individuals can also see

[2] https://eur-lex.europa.eu/eli/reg/2016/679/oj.

[3] Note that "10 cents" is a wrong answer. If the ball costs 10 cents, then the bat will cost 1 euro, so the difference would be 90 cents.

the situation differently and even learn from it. XAI systems can transmit their knowledge and arguments to the users, so if the system is right, then users can learn from it, or in case the AI system is mistaken, users can correct it or update it with the missing information.

2. Preliminaries: the notions of *rationality* and *explanation*

Rationality is a feature commonly included in the diverse theories on intelligence and recently has been introduced as a research topic into experimental and engineering sciences [12]. In the previous section, we argued that trustworthy AI needs rationality, but note that we could add that since rationality is conceived as a crucial part of intelligence, general AI requires addressing the question of rationality. The notion of *rationality* is usually defined as the quality of being based on or following reason. It applies to a wide range of domains and entities [13]. For instance, it is common to say that some emotions, beliefs, societies, reactions, or people themselves are rational. Many different disciplines, such as philosophy, economics, or psychology, have studied rationality from diverse points of view (for a review of the study of this topic, the reader is referred to [13]). Therefore, it is not surprising that rationality has been conceptualized and modeled in different forms. For instance, mainly four kinds of models of rationality have been considered in cognitive science and psychology (see [14] for a general presentation). And the diversity of proposals is even richer in philosophy [13]. Research in AI has been focused on the study of rational agents, and the design of agents often uses the belief-desire-intention model [15] as a rationality paradigm. The utility-maximizing account of rationality, inspired by the instrumental approach from philosophy, has been predominant in AI, even if it is not reasonable to consider an agent acting in this fashion rational [12].

As a complementary approach, this paper focuses on the question of rationality applied to explainable classification algorithms (we refer the reader to [16] for a description of the notion of classification in AI). However, as it occurs with the notion of rationality, there is no canonical definition of explanation. Diverse philosophical approaches conceptualized explanations in different and quite irreconcilable ways. In the literature, an explanation has been defined as a deductive relation, a probabilistic relation, a provision of causes, or a causal relation (we refer the reader to [13, Ch.19]). So, defining the notion of explanation is still a philosophical problem. In this paper, we consider that an *explanation* is any proposition generated by a classification algorithm and labeled by the system itself as such. Next, we can proceed with the principal issue of this work, that is, seeking the criteria that determine whether an explanation of a classification algorithm is rational. For that, we define rational explanations as those meeting certain criteria.

3. Defining *rational explanations*

In this section, we identify, define, and exemplify the criteria of rational explanations in classification algorithms.

3.1. Human understandable

Trivially, the rationality of an explanation can only be analyzed whether the explanation is understandable by people. Instances of explanations satisfying this criterion are: any response of the classical expert system PROSPECTOR written by Richard O. Duda [17] (e.g., *On a scale from -5 to 5, my certainty in MVTD is now: .8995*, or the following answer of the XAI Beer Style Classifier presented in [18]: *It is very likely that this beer is Branche, because its color is pale, its bitterness is low, and its strength is session.*

3.2. Conceptual

Concepts are essential to human thought and play a fundamental function in explaining cognitive tasks as decision-making or categorization [19,20]. Furthermore, according to conceptual coherentism, it is impossible to believe, understand or trust in a proposition (in particular, an explanation) without having the concepts that figure essentially in it [13, Ch.1]. Hence concepts play a very significant role in XAI. Indeed, several philosophical theories on concepts (e.g., connectionism, the analogical approach, or the classical-symbolic one) have inspired diverse AI frameworks like neural networks or formal ontologies [19]. Regarding concepts, we propose three conditions for rational explanations:

- Rational explanations have to use concepts to proceed with the categorization and employ them for building the propositions explaining the result. The explanation *This leaf is not like any known leaf species, but it is round with toothed and lobed margin* [21] meets this condition.
- The concepts used by the explanation must be coherent with human perception, that is, AI systems must align their perception (sensor data) to concepts that people can understand and that they usually use to communicate i.e. in natural language. In the literature, approaches that define concepts that are aligned with human perception are for instance: qualitative descriptors based on reference systems [22], conceptual spaces [23], data-driven conceptual spaces [21], ontologies [24], or fuzzy sets [25]. Note that the mere appearance of concepts is not enough to ensure that the explanations displayed are human understandable. For instance, some machine learning algorithms might build meaningless linguistic labels related to concepts. So, later we must aligned these learned labels with human understanding to gather meaning for the users.
- The concepts used to classify an item into a group have to be related to the distinctive and relevant group's traits. Indeed, an algorithm could correctly classify an item into a category using concepts, and associate them with meaningful linguistic labels, but it could be that these concepts have nothing to do with the group's characteristics within which the item is classified. For instance, shortcuts may fail in real-world scenarios far from standard benchmarks and classify the breed of a dog according, for example, to the presence of the snow in the image. This would lead to an explanation of the form *The dog is a Siberian Husky because it is found in a snowed mountain*, which does not include any dog's traits.

3.3. Context adequate

In linguistics, the goal of Referring Expression Generation (REG) is to find a set of properties (minimal, or psychologically plausible) that are all true for the intended referent, but not all true for any distractor [26,27]. So, in this case, the context of the referent object and the distractor objects must be taken into account.

When referring to an object, multiple categorizations are usually possible [28]. For example, assuming a dog has the following properties: *small*, *black and white*, *tall* and *have spots*, someone may refer to it by saying *the tall black and white dog with spots is a Dalmatian*. However, if that dog is in a context where other dogs are also tall, then including the attribute *tall* in the explanation is not cognitively adequate, i.e., people will not say it since it does not help to distinguish it. If there are several tall dogs and only one dog is small, then we can refer to it by the property that distinguishes it from the rest, i.e., *the small dog is a chihuahua*. In this case, if there are no more small dogs, the color and if it has spots or not is not significant according to the context.

3.4. *Personalized according to users' background*

The rationality of some explanations depend on the user's background, in the sense that one might need specific knowledge to understand them. To exemplify this, let us consider the classification's explanation of a basketball player presented in [29]: *The player is Center because Height is extremely high and Rebounds is medium. There is also a minor chance that it is Small-Forward*. Although the language of this explanation is not the most natural, individuals familiar with basketball may understand it with little effort. However, individuals with no background knowledge in this sport would need an explanation including more suitable terms such as *the player is Center because s/he is the tallest*[4]. Also the notion of *medium* would need more contextualisation, that is, which is the typical number of rebounds in a play, and then what is considered a high/low number of rebound catching for a player. Knowing *high* or *low*, that is the context, we can infer *medium* number of rebounds, but without knowing the thresholds, the meaning of the concept *medium* remains unclear (see Section 3.3 for further details). So, the more adapted to the user's background is an explanation, the more rational the explanation.

3.5. *Coherent with observable human reasoning*

As argued in [30], decision methods must support known patterns of human reasoning. Similarly, we propose another rationality criteria: classification explanations need to be coherent with observable and rational patterns of human reasoning (these patters depend on the classification problem considered). Otherwise, users could not recognize the rationality of the reasoning behind a classification result. For instance, people classify the breed of a dog using reasoning involving its size, its fur, shape, eye type, etc. Let us consider the chihuahua breed, whose average weight ranges [1.8, 2.7] kilograms. Then, if a classification algorithm explained that *This dog is not a chihuahua because its weight is 2.8 kilograms, which is larger than 2.7 kilograms*, users would find this explanation not rational, since it is not the kind of observable human reasoning we do for classifying dogs. A more cognitive statement would include argumental steps like *If a dog is 1.2 meters tall, then this dog cannot ever be classified as a chihuahua*.

Some theoretical tools used to accomplish this objective are logic aggregators (the common idea is to use inputs and outputs of logic aggregators as the conditions filtering those criteria that might serve in mathematical models of human reasoning). For example, when classifying a picture into a painting style, people may have input percepts of the degrees of the adequacy of the image concerning the distinctive traits of the painting

[4] Note that basketball players are usually ordered by height 1–5 and the center is generally number 5.

styles. Humans would aggregate the degrees corresponding to the different features to form a composite percept, assigning a degree of membership of a picture to the painting styles. Based on this theory [30], in [31] the idempotent aggregation, the noncommutativity, and the non-use of annihilators are identified as characteristic patterns of human aggregative reasoning related to the art painting style categorization, and used to explain the results. The idempotent aggregation is based on the assumption that the membership degree to an art style must be between the lowest and the highest value of the traits of this style. The noncommutative reflects that for each painting style, each color trait may have its degree of importance. The explanation classification algorithm presented in [25] did not consider this pattern. An annihilator is an extreme value of suitability (either 0 or 1 – necessary and sufficient condition, respectively) of a feature that is sufficient to decide the result of aggregation regardless of the values of other inputs. An example of the explanations shown in this work is the following one [31]:*The painting is classified in the Post-Impressionism style. The high contrasts between red and green, and between blue and yellow evidence this style*. In that case, the item has been classified taking into account the three patterns of human aggregative reasoning mentioned.

3.6. Contrastive and counterfactual

Factual explanations are based on the features of the input data instances. In contrast, we find contrastive and counterfactual explanations. The conceptual similarity between these two last kinds of explanations motivates us to present them together (see [32] for a detailed introduction to contrastiveness and counterfactuals).

On the one hand, contrastive explanations in classification algorithms give reasons why an item is not classified differently. This kind of explanation describes a classification result by answering the question *Why was the item classified in P rather than in Q?*. In brief, an explanation is contrastive whenever faces the classification result to one of the possible other classifications. Diverse research argues that contrastive explanations are inherent to human cognition [33,34,35], and thus relevant to XAI. In philosophy, contrastive explanations are claimed to be necessary for moral responsibility [36]. And, furthermore, two of the four types of explanatory questions identified by Van Bouwel and Weber [37] lead to contrastive explanations. In light of this, contrastiveness should be a criterion of rational explanations. In [38] the authors propose a method to use questions of this type to restrain the set of features of machine learning algorithms. Often the result is a contrastive explanation, as the following example shows [38]: *System: The flowertype is Setosa. User: Why Setosa and not Versicolor? System: Because for it to be Versicolor the petal width (cm) should be smaller and the sepal width (cm) should be larger*.

On the other hand, counterfactuals picture alternative scenarios to that occurred in fact. Usually, they are presented as conditionals, where the antecedent represents the alternative case, and the consequent describes the consequences of the antecedent. Thus, regarding classification algorithms, counterfactual explanations answer the question *What would have occurred if the item did not hold the property P?*. In short, a counterfactual explanation reveals how the classification could have been different. As Hume pointed out for the first time, counterfactuals explanations play a crucial role in exploring the causes of an event. And more recent approaches from philosophy agree with this [39]. For a cognitive study of counterfactual reasoning, we refer the reader to [40]. Also, some references on ethical aspects related to counterfactual explanations are [41,42,43].

In this way, rational explanations should also include counterfactuals whenever is possible. For instance, the *what-if* tool[5] from *Google* also permits to obtain counterfactual explanations, and in [44] the authors present the method Counterfactual Local Explanations viA Regression (CLEAR). For example, to the question *What if the person does not have a head in the video?*, the XAI model presented in [45] would answer with the counterfactual explanation *Is it possible for a person to exists without the head*. However, as shown in [46] beliefs about additional conditions take precedence over beliefs about presupposed facts for counterfactuals, and thus counterfactuals are not equally helpful in assisting human comprehension. So, future work will consider the criteria established in [46].

4. Case study: explanations on art painting style categorization

In this section, we use the characteristic criteria of rational explainability defined earlier to study the explanations' rationality provided by the art painting style classification algorithm ℓ-SHE [25]. This classifier integrates qualitative descriptors and t-norm based logics and classifies painting from the Baroque, Impressionism, and Post-Impressionism styles using only color features. The main objective of this part is, thus, to exemplify how the research presented in Section 3 might help not only to analyze the explainability of an algorithm but also to improve the rationality of its explanations.

Let us consider the explanations provided by the ℓ-SHE for Renoir's painting *Le djeuner des canotiers* [25]: *rn3 [Le djeuner des canotiers] is an Impressionist painting. The diversity of qualitative colours evidences the Impressionism style. The variety of hues evidences the Impressionism style. The amount of bluish evidences the Impressionism style. The amount of grey evidences the Impressionism style*. Arises the following question: how rational is this explanation? Let us analyze the explanation using the criteria proposed in the previous section. *Is this explanation ...*

- human understandable? It uses natural language and users can understand it.
- conceptual? It utilizes concepts to proceed with the categorization. The concepts (the diversity of color and hues, and the levels of blue and grey) are coherent with human perception. However, color is not the unique distinctive, and relevant trait related to Impressionism; and not considering other features (e.g., the strokes of the picture, or the painting's theme) might not be considered very rational in some classifications.
- context adequate? The ℓ-SHE algorithm selects one or two reasons (from a larger number of facts) to be the explanation, i.e., highlights the more relevant features that characterize the style obtained. Hence we may affirm that the explanation satisfies the criterion.
- personalized according to users' background? Users do not need to be art experts to understand the explanation provided by the system since it is based in color concepts, and color traits that are common sense, that is, most people would understand them. Therefore, the explanation meets this criterion.
- coherent with observable human reasoning? The ℓ-SHE algorithm does not model observable human reasoning behind art painting style classification, as it is indicated in [31]. For example, the categorizations of the styles used in the ℓ-SHE

[5] https://pair-code.github.io/what-if-tool/.

algorithm use t-norms, which admits annihilators. However, art seems too diverse to justify the use of annihilators. For instance, a high level of darkness is distinctive of the Baroque style but other art styles can also present a high level of darkness (e.g., the Romanticism style). Conversely, the absence of this feature is not enough to dismiss this style: some paintings from the Baroque show a few uses of dark colors. Thus, the explanation analyzed does not meet the coherence criteria.

- contrastive and counterfactual? The explanation does not meet the contrastive and counterfactual criteria. This affects the rationality of the explainability of classifications. The accuracy of ℓ-SHE is around 70%. So, often users would ask why an image has been classified into a style rather than another, seeking a contrastive explanation. Or they would wonder what would have been the classification if, for instance, the level of darkness was higher, etc. Therefore, this is a clear issue to improve the explainability of this algorithm.

According to the criteria presented in this paper, we could say that the rationality of the explanation studied is medium. Indeed, it fully meets half of the criteria, two more criteria are few accomplished, and the remaining one is not satisfied. We also remark that this analysis of rationality guides future improvements of the ℓ-SHE algorithm.

5. Conclusions and future work

In this paper, we highlight that rational explanations for AI systems are fundamental. Especially for preserving the users' trust in AI systems and for showing their fairness. We also show that, although rationality is considered as an inherent human feature, on some occasions our thinking might be biased (e.g., ball-and-bat problem).

The main contribution of this paper is the discussion about the criteria that rational explanations must meet. As far as we are concerned, there are no standards for that, but they are very needed. So, this is a step further in that direction. Any explanation produced by an AI system nowadays has been designed by a human. But, as a larger audience would agree, that does not ensure rational explainability. This is why we propose some criteria here in this paper as a guideline to create rational explanations and which are summarized as follows: an explanation produced by an AI system must be human understandable, use concepts, be adequate to the context of communication, be personalized according to the user's background, be contrastive and be coherent with observable human reasoning.

As future work, we intend to explore further the criteria of rational explanations presented in the paper. In addition, we aim to create a dataset showing which explanations in the literature meet the conditions for rationality proposed in this work. Furthermore, we also consider developing an approach for quantifying and qualifying the rationality of explanations provided by classification algorithms. The results obtained by this approach will be compared to the results of a survey carried out to individuals with different profiles (i.e., experts and non-experts in a topic).

Acknowledgments

We thank the referees for their comments. Z. Falomir acknowledges funds from a Ramon y Cajal (RYC2019-027177-I/AEI/10.13039/501100011033) awarded by the MICINN.

References

[1] M. T. Ribeiro, S. Singh, C. Guestrin, "Why should I trust you?": Explaining the predictions of any classifier, in: B. Krishnapuram, M. Shah, A. J. Smola, C. C. Aggarwal, D. Shen, R. Rastogi (Eds.), Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016, ACM, 2016, pp. 1135–1144.

[2] W. Samek, K. Müller, Towards explainable artificial intelligence, in: W. Samek, G. Montavon, A. Vedaldi, L. K. Hansen, K. Müller (Eds.), Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, Vol. 11700 of Lecture Notes in Computer Science, Springer, 2019, pp. 5–22.

[3] H. Hagras, Toward human-understandable, explainable AI, Computer 51 (9) (2018) 28–36.

[4] L. Devillers, F. Fogelman-Soulié, R. Baeza-Yates, AI & human values - inequalities, biases, fairness, nudge, and feedback loops, in: B. Braunschweig, M. Ghallab (Eds.), Reflections on Artificial Intelligence for Humanity, Vol. 12600 of Lecture Notes in Computer Science, Springer, 2021, pp. 76–89.

[5] R. Baeza-Yates, Bias on the web, Commun. ACM 61 (6) (2018) 54–61.

[6] S. Epstein, Integration of the cognitive and the psychodynamic unconscious., American psychologist 49 (8) (1994) 709.

[7] S. A. Sloman, The empirical case for two systems of reasoning, Psychological Bulletin 119 (1996) 3–22.

[8] D. Kahneman, S. Frederick, Representativeness revisited: Attribute substitution in intuitive judgment., in: T. Gilovich, D. Griffin, D. Kahneman (Eds.), Heuristics & Biases: The Psychology of Intuitive Judgment., New York. Cambridge University Press., 2002, pp. 49–81.

[9] D. Kahneman, Thinking, fast and slow, Farrar, Straus and Giroux, New York, 2011.

[10] K. E. Stanovich, R. F. West, Individual differences in reasoning: Implications for the rationality debate?, Behavioral and Brain Sciences 23 (5) (2000) 645665.

[11] S. Frederick, Cognitive reflection and decision making, Journal of Economic perspectives 19 (4) (2005) 25–42.

[12] T. R. Besold, S. L. Uckelman, Normative and descriptive rationality: from nature to artifice and back, J. Exp. Theor. Artif. Intell. 30 (2) (2018) 331–344.

[13] A. Mele, P. Rawling, The Oxford Handbook of Rationality, Oxford Handbooks, Oxford University Press, 2004.

[14] T. R. Besold, Rationality in/for/through ai, in: J. Kelemen, J. Romportl, E. Zackova (Eds.), Beyond artificial intelligence, Vol. 4, Springer, 2013.

[15] M. Bratman, Intention, plans, and practical reason, Harvard University Press, 1987.

[16] M. Fumagalli, G. Bella, F. Giunchiglia, Towards understanding classification and identification, in: A. C. Nayak, A. Sharma (Eds.), PRICAI 2019: Trends in Artificial Intelligence - 16th Pacific Rim International Conference on Artificial Intelligence, Cuvu, Yanuca Island, Fiji, August 26-30, 2019, Proceedings, Part I, Vol. 11670 of Lecture Notes in Computer Science, Springer, 2019, pp. 71–84.

[17] R. O. Duda, S. International., G. S. (U.S.)., N. S. F. (U.S.), Development of the Prospector Consultation System for Mineral Exploration: final report, covering the period October 1, 1976 to September 30, 1978 / by Richard O. Duda ... [et al.], SRI International Menlo Park, Calif, 1978.

[18] J. M. Alonso, A. Ramos-Soto, C. Castiello, C. Mencar, Explainable AI beer style classifier, in: K. Martin, N. Wiratunga, L. S. Smith (Eds.), Proceedings of the SICSA Workshop on Reasoning, Learning and Explainability, Aberdeen, Scotland, UK, June 27, 2018, Vol. 2151 of CEUR Workshop Proceedings, CEUR-WS.org, 2018.

[19] M. Fumagalli, R. Ferrario, Representation of concepts in AI: towards a teleological explanation, in: A. Barton, S. Seppälä, D. Porello (Eds.), Proceedings of the Joint Ontology Workshops 2019 Episode V: The Styrian Autumn of Ontology, Graz, Austria, September 23-25, 2019, Vol. 2518 of CEUR Workshop Proceedings, CEUR-WS.org, 2019.

[20] G. L. Murphy, The Big Book of Concepts, MIT Press, 2002.

[21] H. Banaee, E. Schaffernicht, A. Loutfi, Data-driven conceptual spaces: Creating semantic representations for linguistic descriptions of numerical data, J. Artif. Int. Res. 63 (1) (2018) 691742.

[22] K. D. Forbus, Qualitative modeling, Wiley Interdisciplinary Reviews: Cognitive Science 2 (4) (2011) 374–391.

[23] P. Gärdenfors, Conceptual Spaces, A Bradford Book, 2004.

[24] D. Arvor, M. Belgiu, Z. Falomir, I. Mougenot, L. Durieux, Ontologies to interpret remote sensing images: why do we need them?, GIScience and Remote Sensing 56 (2019) 1–29.

[25] V. Costa, P. Dellunde, Z. Falomir, The logical style painting classifier based on horn clauses and explanations (l-she), Log. J. IGPL 29 (1) (2021) 96–119.

[26] E. Krahmer, K. van Deemter, Computational generation of referring expressions: A survey, Computational Linguistics 38 (1) (2012) 173–218.

[27] R. Dale, E. Reiter, Computational interpretations of the gricean maxims in the generation of referring expressions, Cognitive Science 18 (1995) 233–263.

[28] V. Mast, Z. Falomir, D. Wolter, Probabilistic reference and grounding with pragr for dialogues with robots, Journal of Experimental & Theoretical Artificial Intelligence 28 (5) (2016) 889–911.

[29] J. M. Alonso, Explainable artificial intelligence for kids, in: V. Novák, V. Marík, M. Stepnicka, M. Navara, P. Hurtík (Eds.), Proceedings of the 11th Conference of the European Society for Fuzzy Logic and Technology, EUSFLAT 2019, Prague, Czech Republic, September 9-13, 2019, Vol. 1 of Atlantis Studies in Uncertainty Modelling, Atlantis Press, 2019.

[30] J. Dujmović, Soft Computing Evaluation Logic: The LSP Decision Method and Its Applications, Wiley - IEEE, 2018.

[31] V. Costa, The art painting style classifier based on logic aggregators and qualitative colour descriptors (C-LAD), in: S. Rudolph, G. Marreiros (Eds.), Proceedings of the 9th European Starting AI Researchers' Symposium 2020 co-located with 24th European Conference on Artificial Intelligence (ECAI 2020), Santiago Compostela, Spain, August, 2020, Vol. 2655 of CEUR Workshop Proceedings, CEUR-WS.org, 2020.

[32] I. Stepin, J. M. Alonso, A. Catalá, M. Pereira-Fariña, A survey of contrastive and counterfactual explanation generation methods for explainable artificial intelligence, IEEE Access 9 (2021) 11974–12001.

[33] R. M. J. Byrne, Spatial mental models in counterfactual thinking about what might have been, Trends in Cognitive Sciences 6 (10) (2002) 426–431.

[34] S. Chin-Parker, A. Bradner, A contrastive account of explanation generation, Psychonomic Bulletin & Review 24 (5) (2017) 1387–1397.

[35] T. Miller, Explanation in artificial intelligence: Insights from the social sciences, Artif. Intell. 267 (2019) 1–38.

[36] N. Elzein, The demand for contrastive explanations, Philosophical Studies 176 (5) (2019) 1325–1339.

[37] J. Van Bouwel, E. Weber, Remote causes, bad explanations?, Journal for the Theory of Social Behaviour 32 (4) (2002) 437–449.

[38] J. van der Waa, M. Robeer, J. van Diggelen, M. Brinkhuis, M. A. Neerincx, Contrastive explanations with local foil trees, CoRR abs/1806.07470.

[39] J. Woodward, Making things happen: a theory of causal explanation, Oxford University Press, Oxford, 2003.

[40] R. M. J. Byrne, Cognitive processes in counterfactual thinking about what might have been, Psychology of Learning and Motivation 37 (1997) 105–154.

[41] M. J. Kusner, J. R. Loftus, C. Russell, R. Silva, Counterfactual fairness, in: I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, 2017, pp. 4066–4076.

[42] R. Poyiadzi, K. Sokol, R. Santos-Rodríguez, T. D. Bie, P. A. Flach, FACE: feasible and actionable counterfactual explanations, in: A. N. Markham, J. Powles, T. Walsh, A. L. Washington (Eds.), AIES '20: AAAI/ACM Conference on AI, Ethics, and Society, New York, NY, USA, February 7-8, 2020, ACM, 2020, pp. 344–350.

[43] N. Kilbertus, P. J. Ball, M. J. Kusner, A. Weller, R. Silva, The sensitivity of counterfactual fairness to unmeasured confounding, in: A. Globerson, R. Silva (Eds.), Proceedings of the Thirty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI 2019, Tel Aviv, Israel, July 22-25, 2019, Vol. 115 of Proceedings of Machine Learning Research, AUAI Press, 2019, pp. 616–626.

[44] A. White, A. S. d'Avila Garcez, Measurable counterfactual local explanations for any classifier, in: G. D. Giacomo, A. Catalá, B. Dilkina, M. Milano, S. Barro, A. Bugarín, J. Lang (Eds.), ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020, Vol. 325 of Frontiers in Artificial Intelligence and Applications, IOS Press, 2020, pp. 2529–2535.

[45] A. R. Akula, S. Todorovic, J. Y. Chai, S. Zhu, Natural language interaction with explainable AI models, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019, Long Beach, CA, USA, June 16-20, 2019, Computer Vision Foundation / IEEE, 2019, pp. 87–90.

[46] O. Espino, R. M. J. Byrne, The suppression of inferences from counterfactual conditionals, Cogn. Sci. 44 (4).

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

doi:10.3233/FAIA210166

Applying and Verifying an Explainability Method Based on Policy Graphs in the Context of Reinforcement Learning

Antoni CLIMENT [a], Dmitry GNATYSHAK [b] and Sergio ALVAREZ-NAPAGAO [b]
[a] *Universitat Politècnica de Catalunya, Barcelona, Spain*
[b] *Barcelona Supercomputing Center, Barcelona, Spain*

Abstract. The advancement on explainability techniques is quite relevant in the field of Reinforcement Learning (RL) and its applications can be beneficial for the development of intelligent agents that are understandable by humans and are able cooperate with them. When dealing with Deep RL some approaches already exist in the literature, but a common problem is that it can be tricky to define whether the explanations generated for an agent really reflect the behaviour of the trained agent. In this work we will apply an approach for explainability based on the creation of a Policy Graph (PG) that represents the agent's behaviour. Our main contribution is a way to measure the similarity between the explanations and the agent's behaviour, by building another agent that follows a policy based on the explainability method and comparing the behaviour of both agents.

Keywords. Explainable AI, Reinforcement Learning, Policy Graphs

1. Introduction and Motivation

Humans and complex algorithms for controlling agents do not usually share a common language. In the context of many artificial intelligence methods, including those based on Neural Networks (NN), when evaluating a specific problem we can learn the accuracy, the sensitivity, the reward or the loss with respect to an objective function, along with other metrics. These metrics can give us an idea about whether an algorithm controlling an agent, such as a robot, is learning to perform a certain task correctly or not. However, these metrics are frequently not enough to understand the agent behaviour or the rationale behind their decisions. This is known as the *explainability problem*.

Much of the current state of the art in Reinforcement Learning (RL) is usually enabled by neural network-based methods for finding a (near) optimal policy. This type of programs do not follow a code made by a human that logically solves the problem in a procedural or rule-based manner, but instead they are based on giving training input to generate a non-intelligible complex function and taking the result from it to build a behavioural policy. Furthermore, it is usually difficult to know whether the neural network is learning what it is supposed to, which causes reliability issues.

This type of problem –how can we *explain* the behaviour of an agent and its rationale– is an important sub-field of Explainability of Artificial Intelligence (XAI) [11]. Relevant topics of research in this topic include finding new ways of solving a task, elaborating insights about the agents' strategies, and analyse how an agent takes decisions in specific scenarios where they perform a task better than humans.

The main objective of our work has been to create a graph-based policy in order to analise whether the outcomes of an explainability method are able to accurately describe the behavior of a trained agent. There are currently several approaches to apply explainability methods to RL. In this paper, we briefly overview some of them (Section 2), we choose one based on the creation of graphs representing the observable agent's behaviour, and we apply it to a practical use case (Section 3). In order to validate the results of the method, we present a proposal based on creating a policy based on this generated graph and we use it to find problems on the original proposal as well as to validate the explanations produced (Section 4). The paper ends with a summary of the main conclusions and contributions from the work done (Section 5).

2. State of the art

Over the years, a vast variety of different explainability and interpretability approaches specific to the setting of RL has been proposed. In this section we provide a short overview of some of them and as well as the one that was the foundation of this paper. A more comprehensive and detailed study of the explainability methods in RL can be found in the recent surveys on the topic [1,14].

When considering ways to classify the methods to obtain explanations or describe the inner logic of RL methods, several approaches can be followed. One of the most common of them is to denote whether the explanations are *intrinsic* to the RL model or algorithm itself or are generated *post-hoc*. In addition to that, it is common to further subdivide these approaches by the scope of their explanations into *global* and *local* ones. The former showcase the global strategy used by the agent, while the latter can explain the policy's actions locally on case-by-case basis.

Before we go into specialised approaches, it is important to note that various statistics and metrics can be produced during and after the agent's construction or training, across all of these classifications. For instance, in [15] the authors propose to gather three levels of performance data: the data about the environment, about the behavior of the agent, and the data from the meta-analysis of the previous two, possibly with some domain knowledge. For each level they have outlined a wide array of statistics and metrics that may yield useful insights on agent's performance and the environment's influence on it.

Decision tree models are a classical example of *global intrinsic* approach, with the new methods and their variations still being proposed [16,4]. Here, the RL agent simply needs to "answer" a straightforward series of questions going from the root of a tree-like question structure to get the instructions on which action to choose. Although these methods are meant to be understandable by design, building a decision tree model that can adequately perform in a complex environment usually produces an enormous tree that is too complex to be analysed as whole (thus imposing a trade-off of accuracy and explainability). A number of approaches are proposed to deal with these issues. For instance, [4] proposes to use NN models to generate decision trees, while [16] introduces differentiable decision trees that can be incrementally updated and trained via, for example, gradient descent.

Another example of a *global intrinsic* approach is using agent policies built with some high-level domain-specific programming language [19]. This way the generated policies are transparent by design, you need only to analyse their sequences of commands.

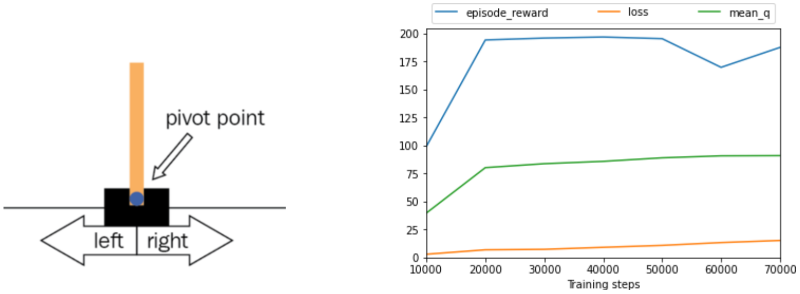
In environments with visual observations we can find a large array of saliency maps-based methods which can represent both *local intrinsic* and *local post-hoc* explainability approaches. For instance, a method proposed in [13] in addition to selecting an action, also generates intrinsic importance maps for the pixels of the image. Alternatively, the classical saliency maps described in [17] use post-hoc backpropagation to find the maps. Additionally, [6] proposes a perturbation-based approach (akin, for example, to the well-known LIME algorithm for deep NN) to generate saliency maps. Although one needs to be careful with utilizing it for some tasks with potentially critical consequences, as perturbation-based methods were recently shown to be prone to adversarial attacks [18].

To get *global post-hoc* explanations, we may analyse the behavior of the trained policy to pinpoint the most interesting or important situations or states, showcasing the behaviour of the agent. Different metrics and approaches can be used to select these execution traces. For instance in [2] the importance measure for states is defined as the difference between the discounted reward values for the best and the worst action choice in this state. Traces centered around the most important states are then shown. Another metric is proposed in [8]: here authors look for critical states which are defined as states for which choosing a random action is significantly worse than choosing a specific one in terms of reward. Further analysis about this family of methods can be found in [3].

Finally, on the border between *global* and *local post-hoc* explanation approaches lie various methods that create simplified representations of the policy or the environment (or its observed version) and then use them to generate local explanations. For instance, in [9,10] authors use NN to generate a graph representation of scenes (images, for instance) that can be later used by a reasoning engine. Another way of doing it is to build a full Markov decision process and traverse it as a graph from the query state to the main reward state [12]. This allows us to ask simple questions about the chosen actions. As an alternative we can simplify the state representation (discretising it if needed) to make the process more feasible in more complex environments [7]. We base our work on the latter approach. It consists of creating a policy graph by creating a mapping from the original state to a set of predicates and then repeatedly running the agent policy, recording its interactions with the environment. This graph of states and actions can then be used for answering simple questions about agent's execution which is shown in Section 3.

3. Generating explanations for the Cartpole scenario

The approach we followed was to attempt to reproduce the method described in [7] with regards to its application on the Cartpole environment. In this paper, the authors present a list of predicates to represent and discretise the states of the environment, and a list of examples of automatically generated explanations, tested against explanations proposed by humans, while accounting for "the occasional presence of incorrect actions taken" by the trained policy. Looking into methodology, we were interested in several aspects of the original work that were not explicitly tested. First of all, we wanted to know how significant the presence of incorrect actions is from a performance point of view and how they might impact the quality of the explanations. Also, we wanted to know whether



(a) Cart Pole standing up (b) Loss, reward and mean q-values of our trained agents

Figure 1. Cart Pole environment and training illustration

there is any method to validate the quality of the explanations complementing human validation.

This environment has been taken from the OpenAI [1] Gym AI library. It consists of a pole attached by an un-actuated joint to a cart, which moves along a frictionless track as shown in Figure 1a. The system is controlled by applying a force going left or right to the cart. The pendulum starts upright and the gravity force will make it fall if not controlled correctly. The goal is to prevent it from falling over. The state is represented with four numbers representing cart position (pos), cart velocity (vel), pole angle (ang) and pole velocity ($velAtT$). It is considered fallen when the pole angle goes beyond 12^o from the center or when the cart goes out of the truck.

Connected to this environment, we have used the *keras-rl2* python library. We created an experimentation pipeline based on training Deep Q Network (*DQN*) Agents using, as in [7], a neural network with 3 convolutional layers with ReLU as an activation function, a reward function equal to the number of steps without the pole fallen and a learning rate of 1e-3. The average loss (based on MAE), episode reward and mean q-values of the training of 75 agents are summarised in Figure 1b.

Our objective was not to find the optimal policy for solving this problem, but rather to find a method to generate agents with variable levels of performance in order to analyse the outcomes of the explainability method at these different levels, and to study the impact of the presence of incorrect actions. From Figure 1b we can see that our agents receive a poor reward at 10k steps but at 20k steps they already solve the problem (which is considered to happen when the average reward is higher than 195). However, at 60k steps the reward starts being unstable while the loss function is always growing. Based on these metrics, we used a maximum of 70k steps for training due to not being seemingly useful to put a higher maximum for our study.

The explainability method proposed in [7] is based on generating a policy graph (PG) representing the states and actions observed by the trained agents acting on random environments. A policy graph G is a tuple $\langle R, N, \varepsilon, \phi \rangle$ where R is a root node representing an initial point for decision making, N is a set of nodes, representing states in our case, ε is a set of directed edges, representing actions in our case, and ϕ is the matrix of transition probabilities for the edges [5]. This formalisation allows to represent any multistage stochastic programming problem, including Markov decision processes as a special case.

[1] https://gym.openai.com

In order to build a policy graph, we need to discretise states and actions. For this, we need to define a set of predicates, which will also be useful when trying to transform the state description into natural language. The initial proposal consisted in using 10 predicates (combining terms and domains):

- $pole\_falling(X)$, with $X = left$ when $[ang < 0 \wedge velAtT < 0]$ and $X = right$ when $[ang > 0 \wedge velAtT > 0]$.
- $pole\_stabilizing(X)$, with $X = left$ when $[ang < 0 \wedge velAtT > 0]$, $X = right$ when $[ang > 0 \wedge velAtT < 0]$.
- $pole\_standing\_up()$ when $[-0.0005 < ang < 0.0005]$.
- $cart\_moving(X)$, with $X = left$ when $[vel < 0]$, $X = right$ when $[vel \geq 0]$.
- $cart\_pos(X)$, with $X = far\_left$ when $[pos \leq -2]$, $X = far\_right$ when $[pos \geq 2]$.
- $cart\_near\_middle()$ when $[-2 < pos < 2]$.

The Gym library allows us to introduce a state and an action and have in return the next state, which we can introduce into the agent that will give us the action to be taken. This cycle allows us to have full knowledge of how the run is going. With this data we have all we need to build the PG graph. We made each agent perform 2k runs of the game of 200 steps each one. We stored all the decisions that the agent took for the reached states, as well as the states visited just after taking the decisions. The created graph shows us the probability of taking a certain action for each state, as well as the probability of reaching a state after taking such action.

The policy graph can then be used to answer three questions that help explaining an agent behaviour: *1) What will you do when you are in x state?*, *2) When do you perform x action?* and *3) Why did not you perform x action in y state?*. The answer to 1) is generated by looking for the most used action in the policy graph from the input state that the user wants to check. In 2), the user inputs an action and the policy graph is used to search for the states where this action has the higher probability to be taken. And in 3), with an action and a state as inputs, the policy graph is used to look for nearby (similar) states to the input state and, for each one, there is a check on whether the contrary action is more likely to be taken, in which case the answer will consist on inferring the difference (in terms of holding predicates) between both states.

While testing the explainability algorithm, we found that not all possible states were reached during the policy graph creation phase. To account for potential edge cases, we introduced a modification in the algorithm to enable searching for nearby (similar) states in the policy graph when querying for an unknown state.

4. Validating the explainability method: creation of a graph-based agent policy

Once the explainability algorithm was implemented and improved, we had access to the generation of natural language explanations as shown in [7]. In that case, the validation is carried out by comparing the generated sentences against sentences written by human experts. One of our objectives was to look for complementary methods that could be automated in order to reduce the dependency to such domain-specific experts.

With the answers from the three questions, it was possible to know if they had more or less sense from a human perspective, but it was not possible to ensure that the answers given had any relationship with the behavior (or even the strategy, if there is any) of the agent, and therefore some kind of validation was missing. However, having a PG built from the observation of the agent's behaviour can give us a powerful tool to work with.

For this validation, we propose the creation of an agent policy inferred from this PG structure, trying to mimic the original trained agent's policy, in order to compare the behaviour of both. One concern related to this proposal is that the PG graph is based on a simplification of the states and the actions (using predicates), and therefore such a policy could also be an over-simplification of a policy that is backed by a deep neural network. However, our aim was not to create equal agents but rather to ensure that the explanations generated are able to reflect the trained agent.

Following this proposal, we implemented, using the Gym API, a policy based on answering the first of the three questions (*What will you do when you are in x state?*) for each current state and using the output to determine the action executed on the environment. Once implemented, we started testing it on the environment in order to check whether the policy was functional.

In this process, we encountered one problem: in a high percentage of runs, the pole ended up falling or going out of track. This brought up two issues: the results were unsatisfactory as the performance of the policy was much lower than the trained agent's average reward; and the behaviour of the policy graph-based policy did not quite reflect the original behaviour. Our hypothesis was that this divergence was caused by a poor state representation with the 10 predicates. By our own observations, the trained agent was able to learn a concept not possible to capture by these predicates, namely the pole being displaced left or right but in a stable position (due to the inertia of the moving cart). We solved this by adding two more predicates to the state representation:

- $stuck(X)$, with $X = left$ when $[\neg pole\_standing\_up() \wedge \neg\exists x : pole\_falling(x) \wedge \neg\exists y : pole\_stabilizing(y) \wedge ang > 0]$, $X = right$ when $[\neg pole\_standing\_up() \wedge \neg\exists x : pole\_falling(x) \wedge \neg\exists y : pole\_stabilizing(y) \wedge ang < 0]$.

By making this change, the new policy inferred from the PG generated using the 12 predicates turned out to be feasible and effective, as we will see later in this section. This issue indicates that human validation might not be enough when dealing with behaviours of agents in scenarios representing complex systems. In order to analyse the behaviour of the policy graph-based agent, we trained 15 agents for a number of steps between 10k and 70k, for a total of 105 agents. For each of these agents, we generated its corresponding policy graph, with the intention of inferring a policy based on answering the first question.

However, doing a direct comparison between both agents can yield misleading conclusions due to the *curse of dimensionality*: because the Cartpole scenario defines a complex system, reducing the state from a high amount of real valued variables to discrete predicates with small domains may entail frequent deviations in the actions taken by each policy in certain states. On the one hand we have a policy that is the product of training with a Q-learning algorithm in a space of states and actions defined by combinations of continuous variables that might be difficult to interpret; on the other hand we have a policy that is based on a very reduced set of predicates that have been designed with interpretability in mind. It cannot be assumed that both policies will have the same expressive power.

However, if we assume that the policy graph is able to generalise (or else the explainability method is pointless), we should be able to ignore these deviations and consider the strategy or general behaviour of the two policies compatible. Our hypothesis is that if we had a single policy that randomly chose actions based on the DQN-policy or the policy graph indistinctly, the behaviour of the agent should stay consistent with respect to the original one. The actions chosen by the policy graph should not interfere –or rather, should be compatible with– the actions chosen by the DQN-policy, as any deviations caused by the divergence between the policies should be mitigated in the long term by the actions chosen by the more fine-grained DQN-policy. To this mixed policy, we also add control policies for checking the behaviour against random agents. Therefore, for each trained agent we generated the following policies:

- DQN: all actions are chosen by the policy trained by the Deep Q-Network.
- PGR: all actions are chosen by the policy inferred from the policy graph.
- RND: each scenario step, the action is chosen at random from all valid actions.
- HEX: each scenario step, the action is chosen at random between DQN and PGR.
- HRD: each scenario step, the action is chosen at random between DQN and RND.

Figure 2 shows a histogram of the last steps the different policies achieved before failing (or succeeding if reaching 200). The two agents with a random selection component, HRD (5.54% success rate) and RND (0.00%) perform very poorly while DQN (78.65%), PGR (78.16%) and HEX (72.95%) have very good and similar success rates. As predicted, the PGR agent success rate deviates considerably at the early steps, as can be seen in the non-marginal frequencies in the histogram between around steps 25 and 50. This can be attributed to the aforementioned deviations due to the state simplification and discretisation, having an impact on the aptitude to stabilise on edge situations, i.e. when the pole is very far from the center. However, DQN and HEX – which is 50% based on PGR – have a very similar histogram, which points at the fact that the PGR policy has not had a strong effect, adjusting well to the original behaviour.

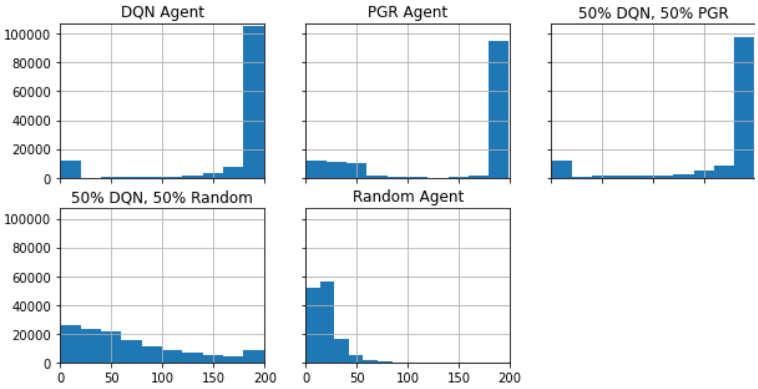


Figure 2. Histogram of last stable step before failing (or succeeding if $step = 200$)

This analysis can be reinforced by analysing the average cart movement, which is a variable that is causally related to the actual behaviour of the agent. In Figure 3 we can see the relationship between the cart movement and the last step before failing or succeeding.

When looking at DQN, we can appreciate two main patterns (ignoring most of the cases, which are successes and they cluster at the right-most border). Across the whole range of steps, most of the failures happen after having moved the cart a low distance (avg.

of around 0.01 and 0.15), forming a wide band from side to side. There seems to be a second pattern coming from those failures in where the cart has moved a higher distance, with the last step not being lower than 100. These patterns indicate the presence of at least two distinct behaviours, which seem to be also distinguishable in HEX and HRD. Again, the effect of the PGR actions in HEX seem to have little effect on the original policy. However, while the patterns seem to be also visible in the PGR case, they are heavily simplified, which may be a consequence of the simplification of the state representation.

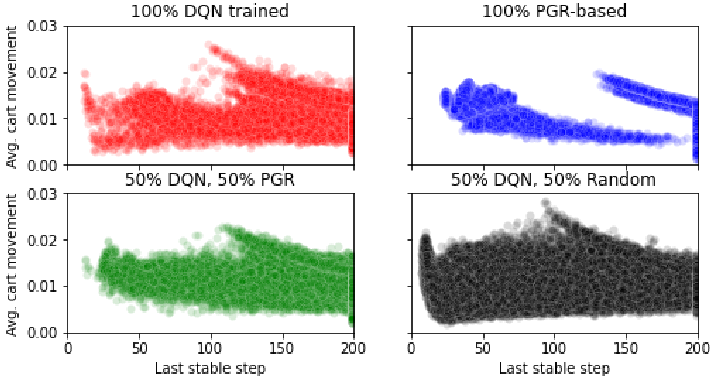


Figure 3. Relationship between last step explored and the average cart movement per step

This raises concerns about the generalisation power of the reduction of the DQN policy into the PGR one. In a more in-depth analysis, we can look at the evolution of the performance based on the amount of training steps. As we saw in Section 3, with our training configuration DQN reaches almost optimal reward between 20k and 50k steps but it becomes unstable after that. In Figure 4 we can see the effect on several metrics: performance, cart movement, pole velocity and pole rotation. The performance of PGR between 10k and 40k steps is above 80% while the performance of DQN is always below 80%, which means that the PGR graph is capable of generalising well until 50k steps.

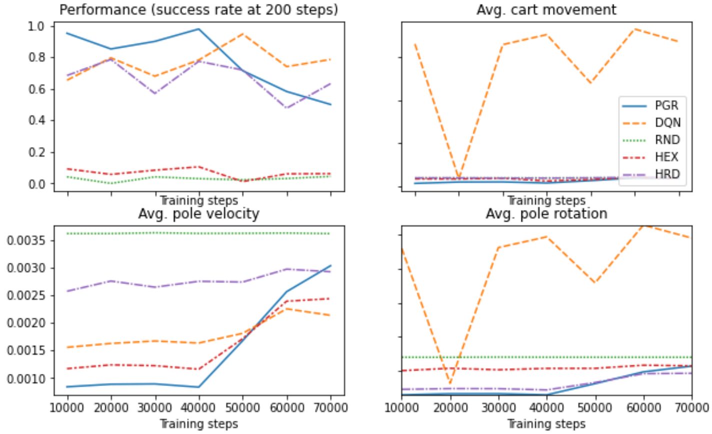


Figure 4. Evolution of metrics through training steps

Finally, we analysed the correlations between the three Cartpole-specific metrics (Figure 5), using Spearman due to these metrics being ordered (all the policies were run on the same random scenarios) but not following a normal distribution. The metric more causally connected to the behaviour (the actions) of the agent, the cart movement, does not entail very high correlations. The highest values are between DQN and HEX (0.60, p<0.001) and between PGR and HEX (0.53, p<0.001), which makes sense as this policy is a combination of both. The correlation between DQN and PGR (0.26, p<0.001) is statistically significant but it is quite low. If we look at the effect of the actions on the pole (both velocity and rotation) we can find higher correlations: DQN and PGR (0.55, p<0.001), DQN and HEX (0.72, p<0.001), PGR and HEX (0.67, p<0.001).

In summary, from the performance results combined with these correlations we can infer that if we look at all the runs globally, the behaviours of the two agents might yield similar results, both in performance and in the effect of the actions (the behaviour of the pole). However, if we look at the individual runs, we will find many frequent deviations. In other words: by applying the explainability method on the Cartpole scenario we can extract rough explanations that can approximate the behaviour or strategy of the original agent *in general*, but the method might not be able to explain with precision the behaviour in every instance of the scenario.

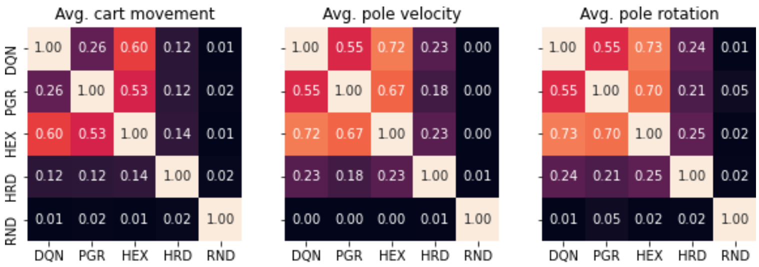


Figure 5. Cross-correlations between cart movement, pole velocity and pole rotation, averaged by step

5. Conclusions

The topic of XAI applied to Reinforcement Learning is growing in relevance and can be key to tackle issues such as the possibility to assess the quality of the behaviour of an agent or aid in the interaction between humans and AI-based agents. There are already some approaches in the literature that try to provide explainability in this context. However, they need to be tested in practical use cases in order to assess their effectiveness.

In this paper, we show our work in this direction in which we choose a method and try to reach a baseline for further research. After an analysis of the literature, we chose a method based on the generation of PGs by discretising the state representation into predicates, and applied it to a simple yet complex scenario (Cartpole). The result was a policy graph that allowed us to produce explanations, that according to the original work should be validated by human experts with domain-specific knowledge.

In order to understand how good the baseline that we were getting was and to be able to validate the results inferred from the policy graph, we propose a method for extending this validation with an automatic process by creating several policies based on both the original policy and the policy graph, and testing them along with the original policy in random new scenarios. By applying this method we were able to 1) detect predicates that were missing in order to have a complete state representation, and 2) analyze the quality of the policy graph as a tool to store a simplified representation of the original behaviour.

This paper presents part of our ongoing work, which is currently advancing in two research lines. First of all, we are extending the generation of the graph-based policy, in order to include not only one but the three types of questions that [7] defines for generating explanations. On a second topic, we are applying the same method to other environments with a much more complex state representation to check whether this method can be generalised. Currently we are applying it to the VizDoom environment.

References

[1] A. Alharin, T. N. Doan, and M. Sartipi. Reinforcement learning interpretation methods: A survey. 8:171058–171077, 2020.

[2] D. Amir and O. Amir. HIGHLIGHTS: Summarizing agent behavior to people. AAMAS '18, page 1168–1176, Richland, SC, 1 2018. IFAAMAS.

[3] Ofra Amir, Finale Doshi-Velez, and David Sarne. Summarizing agent strategies. Autonomous Agents and Multi-Agent Systems, 7 2019.

[4] O. Bastani, Y. Pu, and A. Solar-Lezama. Verifiable reinforcement learning via policy extraction. In Proceedings of the 32nd International Conference on NIPS, NIPS'18, page 2499–2509, Red Hook, NY, USA, 2018. Curran Associates Inc.

[5] O. Dowson. The policy graph decomposition of multistage stochastic programming problems. Networks, 76(1):3–23, 2020. Publisher: Wiley Online Library.

[6] S. Greydanus, A. Koul, J. Dodge, and A. Fern. Visualizing and understanding Atari agents. In Proceedings of the 35th ICML, volume 80 of Proc. of ML Research, pages 1792–1801. PMLR, 10–15 Jul 2018.

[7] B. Hayes and J. A. Shah. Improving robot controller transparency through autonomous policy explanation. In Proceedings of the 2017 ACM/IEEE Intl. Conf. on HRI, page 303–312, NY, USA, 1 2017. ACM.

[8] S. H. Huang, K. Bhatia, P. Abbeel, and A. D. Dragan. Establishing appropriate trust via critical states. 2018 IEEE/RSJ IROS, pages 3929–3936, 2018.

[9] M. Klawonn and E. Heim. Generating triples with adversarial networks for scene graph construction. CoRR, abs/1802.02598, 2018.

[10] M. Klawonn, E. Heim, and J. A. Hendler. Exploiting class learnability in noisy data. CoRR, abs/1811.06524, 2018.

[11] L. Longo, R. Goebel, F. Lecue, P. Kieseberg, and A. Holzinger. Explainable artificial intelligence. In Machine Learning and Knowledge Extraction, pages 1–16, Cham, 2020. Springer.

[12] P. Madumal, T. Miller, L. Sonenberg, and F. Vetere. Explainable reinforcement learning through a causal lens. Proceedings of the AAAI Conference, 34(03):2493–2500, Apr. 2020.

[13] D. Nikulin, A. Ianina, V. Aliev, and S. Nikolenko. Free-lunch saliency via attention in atari agents. In 2019 IEEE/CVF ICCVW, pages 4240–4249, 2019.

[14] E. Puiutta and E. M. S. P. Veith. Explainable reinforcement learning: A survey. In Machine Learning and Knowledge Extraction, pages 77–95, Cham, 2020. Springer.

[15] P. Sequeira, E. Yeh, and M. T Gervasio. Interestingness elements for explainable reinforcement learning through introspection. In IUI Workshops, 1 2019.

[16] A. Silva, M. Gombolay, T. Killian, I. Jimenez, and S.-H. Son. Optimization methods for interpretable differentiable decision trees applied to reinforcement learning. In Proceedings of the 23rd Intl. Conf. on AI and Statistics, volume 108 of Proceedings of ML Research, pages 1855–1865. PMLR, Aug 2020.

[17] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In Proc. of ICLR, 2014.

[18] D. Slack, S. Hilgard, E. Jia, S. Singh, and H. Lakkaraju. Fooling LIME and SHAP: Adversarial attacks on post hoc explanation methods. pages 180—-186, 1 2020.

[19] A. Verma, V. Murali, R. Singh, P. Kohli, and S. Chaudhuri. Programmatically interpretable reinforcement learning. CoRR, abs/1804.02477, 2018.

Subject Index

This page intentionally left blank

Artificial Intelligence Research and Development
M. Villaret et al. (Eds.)

Author Index