SACAIR 2020

Proceedings of the

First Southern African Conference for

Artificial Intelligence Research

December 2020

AI
SACAIR 2020
Southern African Conference
for Artificial Intelligence Research
22-26 February 2021

Editor: Aurona Gerber

Department of Informatics
University of Pretoria

Copyright Notice

The SACAIR 2020 organising committee is not responsible for errors or omissions from individual papers contained in these proceedings. Technical electronic anomalies are possible and unavoidable during the compilation process. The publisher is not responsible for the accuracy and validity of information contained in these papers, nor is it responsible for the final use to which this book may be used.

The SACAIR 2020 Proceedings Editors attest as follows: All conference paper submissions that appear in these proceedings have been through a double-blind peer review process prior to acceptance into the final conference programme.

Editor: Aurona J. Gerber, Department of Informatics, University of Pretoria

Published by the SACAIR 2020 Organising Committee
Center for AI Research (CAIR)
Private Bag X20
Hatfield
0028
ISBN: 978-0-620-89373-2(e-book)

Preface

This volume contains the revised accepted papers of SACAIR 2020, the *1st Southern African Conference for Artificial Intelligence Research*[1]. A selection of the best papers were published in a volume of Springer CCIS (CCIS 1342 - available at https://www.springer.com/gp/book/9783030661502), and for these papers only the abstracts are included in this volume as the final published papers appear in the Springer publication.

Foreword from the Conference Chair

Dear authors and readers,

It is with great pleasure that I write this foreword to the Proceedings of the first Southern African Conference for Artificial Intelligence Research (SACAIR 2020), to be held on the West Rand of Johannesburg, South Africa, on 22 to 26 February 2021[2]. The programme includes an unconference for students on 22 February (a student driven event for students to interact with each other as well as with sponsors and other possible employers), a day of tutorials on 23 February and the main conference from 24-26 February.

SACAIR 2020 is the second international conference focussed on Artificial Intelligence hosted by the Centre for AI Research (CAIR), South Africa. The inaugural CAIR conference, the Forum for AI Research (FAIR 2019) was held in Cape Town, South Africa, in December 2019, and SACAIR 2020 will build on its success.

The Centre for AI Research (CAIR)[3] is a South African distributed research network that was established in 2011 with the aim of building world class Artificial Intelligence research capacity in South Africa. CAIR conducts foundational, directed and applied research into various aspects of AI through its nine research groups based at six universities (the University of Pretoria, the University of KwaZulu-Natal, the University of Cape Town, Stellenbosch University, University of the Western Cape and North West University). Research groups at CAIR include an Adaptive and Cognitive Systems Lab situated at the University of Cape Town, an AI and Cybersecurity research group at the University of the Western Cape, an AI for Development and Innovation group at the University of Pretoria, two Machine Learning groups focused on deep learning at North West University and the University of Kwa-Zulu Natal, a Knowledge Abstraction and Representation group at Stellenbosch University, an Ethics of AI research group at the University of Pretoria, a Knowledge Representation and Reasoning group at the University of Cape Town, and a Mathematical and Computational Statistics group focused on applied data science at the University of Pretoria.

[1] https://sacair.org.za/

[2] The original date was 30 November – 4 December 2020, but, due to the COVID-19 pandemic, the conference was pushed into 2021 in the hope of being able to retain its face-to-face format in the interest of building an AI community in South Africa.

[3] https://www.cair.org.za/

The theme for SACAIR 2020 is *AI transforming Humanity.* AI technologies in their current data-driven form have the potential to transform our world for the better. Applications of AI technologies in healthcare, agriculture, restoration of the environment and ecosystems, energy, water management, identification of social patterns and bias, law enforcement, education, information, connectivity, smart city and infrastructure planning, performing and creative arts, and many other areas are proof of this.

However, humans are faced with serious challenges in the context of AI advances in all areas of their lives, as wide apart as employment and labour on the one hand and social companionship on the other. In the context of machine learning applications, these challenges lead to concerns around fairness, structural bias and amplification of existing social stereotypes, privacy, transparency, accountability and responsibility, and trade-offs among all these concerns, especially within the context of security, robustness and accuracy of AI systems. Furthermore, AI technologies can perform tasks that previously only humans could perform, such as calculating the best treatment for certain illnesses and caring for older persons. In some cases this is a good thing, but in some it challenges human agency and experience, and even political stability in profound ways. Human notions of morality, of responsibility, and of ethical decision-making are challenged in ways humanity has never before encountered. In addition, children grow up in novel contexts impacted on by technological manipulation of social narratives and we do not yet know what the impact of this will be. In its turn, media and information literacy has become an essential skill just as important as technical skills. Finally, there are also cultural concerns such as the loss of nuances of human languages and expression in the context of NLP, concerns around the ownership of art, and others.

The choice of conference theme was intended to ensure multi-disciplinary contributions that focus both on the technical aspects and social impact and consequences of AI technologies. In addition, there is a healthy balance between contributions from logic-based AI and those from data-driven AI, as the focus on knowledge representation and reasoning remains an important ingredient of studying and extending human intelligence. In line with the above, it was decided that the conference topics would cover five broad areas of Artificial Intelligence: Machine Learning, Knowledge Representation and Reasoning, Applications of AI, AI for Ethics and Society, and AI for Development and Social Good. In line with the theme, Peter-Paul Verbeek, the chair of the UNESCO Commission on the Ethics of Scientific Knowledge and Technology (COMEST), and chair of the Philosophy of Human-Technology Relations research group and co-director of the DesignLab at the University of Twente (Netherlands), will deliver the opening keynote.

We expect this multi- and interdisciplinary conference to grow into the premier AI conference in Southern Africa as it brings together nationally and internationally established and emerging researchers from across various disciplines including Computer Science, Mathematics, Statistics, Informatics, Philosophy and Law. The conference is also focused on cultivating and establishing a network of talented students working in AI from across Africa.

I sincerely thank the technical chair, Aurona Gerber, for her hard work on the volume and the editorial duties performed. A thank you to the programme chairs (Aurona Gerber, Anne Gerdes, Giovanni Casini, Marelie Davel, Alta de Waal, Anban Pillay, Deshendran Moodley and Sunet Eybers), the local and international panel of reviewers, our keynotes and the authors and participants for their contributions. Last but not least, our gratitude to the members of the organising committee (Aurona Gerber, Anban Pillay, Alta de Waal), student organisers (Karabo Maiyane, Emile Engelbrecht, Nirvana Pillay and Yüvika Singh) and our sponsors, specifically the AIJ division of IJCAI, without whom this conference would not have been realised.

November 2020

Emma Ruttkamp-Bloem
Organising Chair: SACAIR 2020

Message from the Technical Chair

Dear readers,

This volume of CCIS contains the revised accepted papers of SACAIR 2020. We are thankful that our first annual Southern African Conference for Artificial Intelligence Research elicited the support it did during this challenging year with all the uncertainties due to the Covid-19 pandemic.

We received more than 70 abstracts, and after submission and a first round of evaluation, 53 papers were sent out for review to our SACAIR programme committee. The 53 SACAIR submissions were solicited according to five topics: AI for Ethics and Society (9), AI in Information Systems, AI for Development and Social Good (3), Applications of AI (25), Knowledge Representation and Reasoning (8), Machine Learning Theory (8).

The programme committee comprised 72 members, 13 of whom were from outside Southern Africa. Each paper was reviewed by at least three members of the programme committee in a rigorous, double-blind process whereby especially the following criteria were taken into consideration: Relevance to SACAIR, Significance, Technical Quality, Scholarship, and Presentation that included quality and clarity of writing. For this SACAIR online proceedings volume, 19 full research papers were selected for publication in Springer (which translates to an acceptance rate of 35.8%) whilst a further 20 papers were accepted for inclusion in this volume, which amounts to an acceptance rate of 55% for all submissions, or 73% for reviewed submissions. The accepted full research papers per topic are: AI for Ethics and Society (6), AI in Information Systems, AI for Development and Social Good (2), Applications of AI (17), Knowledge Representation and Reasoning (7), and Machine Learning Theory (7).

Thank you to all the authors and programme committee members, and congratulations to the authors whose work was accepted for publication in this Springer volume. We wish our readers a fruitful reading experience with these proceedings!

December 2020

Aurona Gerber
Technical Chair: SACAIR 2020

SACAIR Sponsors

The sponsors of SACAIR 2020, *The Journal of Artificial Intelligence* and the *Centre for AI Research (CAIR)*, are herewith gratefully acknowledged.

Organisation

AI
SACAIR 2020
Southern African Conference
for Artificial Intelligence Research
22-26 February 2021

General Chair

Emma Ruttkamp-Bloem	University of Pretoria, South Africa & Centre of AI Research (CAIR), South Africa

Program Committee Chairs

Technical Chair

Aurona Gerber	University of Pretoria, South Africa & Centre of AI Research (CAIR), South Africa

Topic Chairs: Applications of AI

Anban Pillay	University of Kwazulu-Natal, South Africa & Centre of AI Research (CAIR), South Africa
Deshendran Moodley	University of Cape Town, South Africa & Centre of AI Research (CAIR), South Africa

Topic Chairs: AI for Ethics and Society

Emma Ruttkamp-Bloem	University of Pretoria, South Africa & Centre of AI Research (CAIR), South Africa
Anne Gerdes	University of Southern Denmark, Denmark

Topic Chairs: AI in Information Systems

Aurona Gerber	University of Pretoria, South Africa & Centre of AI Research (CAIR), South Africa
Sunet Eybers	University of Pretoria, South Africa

Topic Chair: Knowledge Representation and Reasoning

Giovanni Casini	ISTI - CNR, Italy & University of Cape Town, South Africa

Topic Chair: Machine Learning Theory

Alta de Waal	University of Pretoria, South Africa & Centre of AI Research (CAIR), South Africa
Marelie Davel	North-West University, South Africa. Centre of AI Research (CAIR), South Africa

Local Organising Committee

Anban Pillay	University of Kwazulu-Natal, South Africa
Alta de Waal	University of Pretoria, South Africa
Julie-Anne Sewparsad	University of Pretoria, South Africa
Karabo Maiyane	University of Pretoria, South Africa
Emile Engelbrecht	Stellenbosch University, South Africa
Renee le Roux	Mongoose Communications & Design, South Africa

Programme Committee

Etienne Barnard	North-West University, South Africa
Sihem Belabbes	LIASD, Université Paris, France
Sonia Berman	University of Cape Town, South Africa
Jacques Beukes	North-West University, South Africa
Willie Brink	Stellenbosch University, South Africa
Arina Britz	Stellenbosch University, South Africa
Michael Burke	University of Edinburgh, United Kingdom
Jan Buys	University of Cape Town, South Africa
Giovanni Casini	ISTI - CNR, Italy
Colin Chibaya	Sol Plaatje University, South Africa
Olawande Daramola	Cape Peninsula University of Technology, South Africa
Jérémie Dauphin	University of Luxembourg, South Africa

Marelie Davel	North-West University, South Africa
Tanya de Villiers Botha	Stellenbosch University, South Africa
Alta De Waal	University of Pretoria, South Africa
Febe de Wet	Stellenbosch University, South Africa
Iena Derks	University of Pretoria, South Africa
Tiny Du Toit	North-West University, South Africa
Andries Engelbrecht	University of Stellenbosch, South Africa
Sunet Eybers	University of Pretoria, South Africa
Inger Fabris-Rotelli	University of Pretoria, South Africa
Sebastian Feld	Ludwig Maximilian University of Munich, Germany
Eduardo Fermé	Universidade da Madeira, Portugal
Anne Gerdes	University of Denmark, Denmark
Mandlenkosi Gwetu	University of KwaZulu-Natal, South Africa
Shohreh Haddadan	University of Luxembourg, Luxembourg
Henriette Harmse	EMBL-EBI, United Kingdom
Michael Harrison	University of Cape Town, South Africa
Bertram Haskins	Nelson Mandela University, South Africa
Marie Hattingh	University of Pretoria, South Africa
Omowunmi Isafiade	University of the Western Cape, South Africa
Edgar Jembere	University of KwaZulu-Natal, South Africa
Herman Kamper	Stellenbosch University, South Africa
Lisa Kirkland	University of Pretoria, South Africa
Eduan Kotzé	University of the Free State, South Africa
Jaco Kruger	St Augustine College of South Africa, South Africa
Louise Leenen	University of the Western Cape, South Africa
Aby Louw	CSIR, South Africa
Patricia Lutu	University of Pretoria, South Africa
Patrick Marais	University of Cape Town, South Africa
Vukosi Marivate	University of Pretoria, South Africa
Réka Markovich	University of Luxembourg, Luxembourg
Muthoni Masinde	Central University of Technology, South Africa
Jocelyn Mazarura	University of Pretoria, South Africa
Felix McGregor	Saigen, South Africa
Thomas Meyer	University of Cape Town, South Africa
Deshendran Moodley	University of Cape Town, South Africa
Vincent C. Müller	University of Leeds, United Kingdom
Peeter Müürsepp	University of Talinn, Estonia
Fred Nicolls	University of Cape Town, South Africa
Geoff Nitshcke	University of Cape Town, South Africa
Oluwafemi Oriola	University of the Free State, South Africa
Anban Pillay	University of Kwazulu-Natal, South Africa
Arnold Pretorius	North West University, South Africa
Laurette Pretorius	University of South Africa, South Africa
Catherine Price	University of KwaZulu-Natal, South Africa
Helen Robertson	University of the Witwatersrand, South Africa

Irene Russo	CNR Pisa, Italy
Emma Ruttkamp-Bloem	University of Pretoria, South Africa
Jonathan Shock	University of Cape Town, South Africa
Riana Steyn	University of Pretoria, South Africa
Umberto Straccia	ISTI-CNR, Italy
Jules-Raymond Tapamo	University of KwaZulu-Natal, South Africa
Anitta Thomas	University of South Africa, South Africa
Dustin Van Der Haar	University of Johannesburg, South Africa
Terence Van Zyl	University of Johannesburg, South Africa
Peter-Paul Verbeek	University of Twente, Netherlands
Serestina Viriri	University of KwaZulu-Natal, South Africa
Bruce Watson	Stellenbosch University, South Africa
Adrian Weller	University of Cambridge, United Kingdom

Table of Contents

Part I

AI for Ethics and Society

AI for Ethics and Society: Abstracts of Full Papers Published in Springer CCIS Volume 1342

The papers in this section appear in Springer CCIS Volume 1342
available at https://link.springer.com/book/10.1007/978-3-030-66151-9.
The abstracts are included in this proceedings.

- Friedman, Cindy. *Human-Robot Moral Relations: Human Interactants as Moral Patients of their Own Agential Moral Actions Towards Robots.*
- Kruger, Jaco. *Nature, culture, AI and the common good – Considering AI's place in Bruno Latour's Politics of Nature.*
- Ruttkamp-Bloem, Emma. *The Quest for Actionable AI Ethics.*

Human-Robot Moral Relations: Human Interactants as Moral Patients of their Own Agential Moral Actions Towards Robots

Cindy Friedman[1, 2] [0000-0002-4901-9680]

[1] Department of Philosophy, University of Pretoria, Pretoria, South Africa
cindzfriedman@gmail.com
[2] Centre for AI Research (CAIR), South Africa

Abstract. This paper contributes to the debate in the ethics of social robots on how or whether to treat social robots morally by way of considering a novel perspective on the moral relations between human interactants and social robots. This perspective is significant as it allows us to circumnavigate debates about the (im)possibility of robot consciousness and moral patiency (debates which often slow down discussion on the ethics of HRI), thus allowing us to address actual and urgent current ethical issues in relation to human-robot interaction.The paper considers the different ways in which human interactants may be moral patients in the context of interaction with social robots: robots as conduits of human moral action towards human moral patients; humans as moral patients to the actions of robots; and human interactants as moral patients of their own agential moral actions towards social robots. This third perspective is the focal point of the paper. The argument is that due to *perceived* robot consciousness, and the possibility that the immoral treatment of social robots may morally harm human interactants, there is a unique moral relation between humans and social robots wherein human interactants are both the moral agents of their actions towards robots, as well as the *actual* moral patients of those agential moral actions towards. Robots, however, are no more than *perceived* moral patients. This discussion further adds to debates in the context of robot moral status, and the consideration of the moral treatment of robots in the context of human-robot interaction.

Keywords: Robot Ethics, Human-Robot Interaction, Moral Patiency.

This paper is published in the Springer CCIS volume at https://link.springer.com/chapter/10.1007/978-3-030-66151-9_1

Nature, Culture, AI and the Common Good – Considering AI's Place in Bruno Latour's *Politics of Nature*

Jaco Kruger [0000-0002-7785-7374]

[1]St Augustine College of South Africa, Johannesburg, South Africa
[2]Faculty of Theology, North West University, Potchefstroom, South Africa
j.kruger@staugustine.ac.za

Abstract. This paper considers the place and the role of AI in the pursuit of the common good. The notion of the common good has a long and venerable history in social philosophy, but this notion, so it is argued, becomes problematic with the imminent advent of Artificial General Intelligence. Should AI be regarded as being in the service of the common good of humanity, or should the definition of the social common rather be enlarged to include non-human entities in general, and AI's, which in the future may include human level and superhuman level AI's, in particular? The paper aims to clarify the questions and the concepts involved by interpreting Bruno Latour's proposal for a politics of nature with specific reference to the challenge posed by the imminent advent of human level artificial general intelligence (AGI). The recent suggestion by eminent AI researcher, Stuart Russell, that the pursuit of AI should be re-oriented towards AI that remain in the service of the human good, will be used as a critical interlocutor of Latour's model. The paper concludes with the suggestion that the challenge will be to steer a middle ground between two unacceptable extremes. On the one hand the extreme of a "truth politics" that assumes there is a pure human nature and definite human interests that must be protected against AI should be avoided. On the other hand, the alternative extreme of a naked "power politics" must also be avoided because there is a very real possibility that super AI may emerge victorious out of such a power struggle.

Keywords: Bruno Latour • Artificial Intelligence • Common Good • Stuart Russell.

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_2

The Quest for Actionable AI Ethics

Emma Ruttkamp-Bloem[1,2][0000−0003−0299−6406]

[1] Department of Philosophy, University of Pretoria, Pretoria, South Africa
emma.ruttkamp-bloem@up.ac.za,
[2] Centre for AI Research (CAIR), South Africa

Abstract. In the face of the fact that AI ethics guidelines currently, on the whole, seem to have no significant impact on AI practices, the quest of AI ethics to ensure trustworthy AI is in danger of becoming nothing more than a nice ideal. Serious work is to be done to ensure AI ethics guidelines are actionable. To this end, in this paper, I argue that AI ethics should be approached 1) in a multi-disciplinary manner focused on concrete research in the discipline of the ethics of AI and 2) as a dynamic system on the basis of virtue ethics in order to work towards enabling all AI actors to take responsibility for their own actions and to hold others accountable for theirs. In conclusion, the paper emphasises the importance of understanding AI ethics as playing out on a continuum of interconnected interests across academia, civil society, public policy-making and the private sector, and a novel notion of 'AI ethics capital' is put on the table as outcome of actionable AI ethics and essential ingredient for sustainable trustworthy AI.

Keywords: AI Ethics, Virtue Ethics, Multi-disciplinary Research, AI Ethics Capital, Trustworthy AI.

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_3

AI for Ethics and Society: Full Papers Accepted for SACAIR 2020 Online Proceedings

The following full papers are included in this proceedings. These papers can be cited as indicated below adding page numbers and the url to the specific paper.

- Breytenbach, Johan and Van den Berg, Carolien. *Embedding Ethics into 4IR Information Systems Teaching and Learning.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
- Combrink, Herkulaas; Marivate, Vukosi and Rosman, Benjamin. *A Framework for Undergraduate Data Collection Strategies for Student Support Recommendation Systems in Higher Education.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
- Rufus, Heather. *The Implications of Personalization Algorithms on Individual Autonomy.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.

Embedding Ethics into 4IR Information Systems Teaching and Learning

Research in Progress

Johan Breytenbach[1[0000-0001-7883-7140]] and Carolien van den Berg[2[0000-0002-2243-8375]]

[1] University of the Western Cape, Cape Town, South Africa
jbreytenbach@uwc.ac.za
[2] University of the Western Cape, Cape Town, South Africa
cvandenberg@uwc.ac.za

Abstract. As the data analytics (DA), machine learning (ML), artificial intelligence (AI) and data modelling (DM) components of the international Information Systems (IS) curricula mature, there is a growing concern regarding the complexity of teaching ethics within the current IS curriculum. This paper investigates ethics as it relates to 4IR technologies, with the purpose of proposing core content to be taught to IS students at the graduate level, as well as a set of curriculum design considerations. There is a need for an IS curriculum that equips IS students to design ethical ML and AI systems, and teachers equipped to teach such a curriculum. Surveying IS graduate-level students' understanding and perceptions of ethics as it relates to emerging technologies before receiving tuition in key competency areas provides insights into the structure and content required to embed ethics into IS courses. This research has implications for higher education institutions that are offering data analytics, machine learning, and/or artificial intelligence courses as part of IS degrees. Key findings include students' inability to correctly diagnose ethical considerations, and their varied understanding of the responsibilities involved in ensuring ethical design and usage of AI and related technologies. Students' understanding of the management and assignment of responsibility throughout the system design process also receive attention. This paper responds to a strong call from extant literature for ethical leadership and ethical sensitivity within Information Systems design processes, resulting in a curriculum that promotes the design of systems that are "ethical-by-design".

Keywords: Artificial Intelligence, Ethics, IS Education, 4IR

1 Introduction and Background

The 4th Industrial Revolution (4IR) is typified as a dynamic era of rapid change, complexity, fluidity and ubiquitous technology within all realms of our everyday existence. Technology such as machine learning (ML), artificial intelligence (AI), algorithms, big data, automation and robotics are currently being applied in a wide range of fields, and designers are just beginning to understand the implications of these developments for design practice [1]. The impact of these technologies on society, communities and individuals is presenting challenges to the traditional conceptualisation of design as a human-focused endeavour [2] and often exceeds the human capacity to adapt [3].

This changing world of work requires IS graduates with the capability to understand their role as designers of future technologies, how the technologies they design work and the ethical, legal, and commercial controversies these developments are calling into question [4]. In this article, we focus on how ethics may be incorporated into the 4IR system design classroom, suggest content guidelines, and provide guidance on the teaching and learning of ethics in the 4IR system design classroom in the form of curriculum design principles.

Literature about ethics within IS as it relates to 4IR technologies is still maturing. Stahl [5] is one of the first to present a framework for the classification of ethics theory within IS, providing a valuable distinction between moral intuition, explicit morality, ethical justification, and higher-level reflexivity. Stahl's view of the difficulty of teaching ethical and moral reflection to IS students with little experience in the fields of ethics and morality echo's the experiences of the authors of this paper, "[A]chieving this is a tall order and goes in many ways beyond what we currently expect students of IS, or most other subjects, to achieve" [5, p258].

It is the view of the authors that 4IR technologies present IS students with ethical questions more frequently than in the past. System design and implementation now frequently result in reflections on the role of human actors, the role of machines, automated decision making, and the impact of these technologies on the health, privacy, and security of individuals and society. Floridi et al., [6, p705) presents a unified framework of ethical considerations around the design and use of AI, and makes a call as leading academic in the field of AI ethics for, "A European-level recommendation to include ethics and human rights in the degrees of data and AI scientists and other scientific and engineering curricula dealing with computational and AI systems". This article is a first step towards heeding the call within an African context.

This paper summarises ethics as a knowledge domain from existing literature and presents a list of key ethics theories and competencies to be taught to IS (design) students at the graduate level. The teaching and learning aspects related to bringing the large and complex field of ethics into the IS curriculum are also considered and reduced to curriculum design principles. The article then offers its primary contribution: a suggestion for how to design an AI ethics course that is approachable for IS graduates and prepares participating students for ethically healthy careers in system design and data analytics. This contribution is timely, as the revised IS 2020 curriculum suggestions from the ACM (due for publication end 2020) highlights ethics as a new key focus area of the IS curriculum [7].

2 Research Problem

Embedding ethics into the graduate level IS classroom is a challenge faced by many IS educators. This article aims to further the teaching of ethics in IS - as it relates to 4IR technologies – by doing the following:

- Giving an overview, and guidance from literature and experience, of basic ethics concepts that should be considered for inclusion into a graduate-level, IS course on ethics.
- Suggesting design principles for the design of an IS graduate-level course in ethics

Findings presented in this article are based mainly on data gathered by surveying graduate-level IS students before receiving lectures on ethics as it relates to IS.

Graduate-level students in IS, at a single university in South Africa, were tasked with ranking different definitions and outcomes relating to ethics theory and ethical scenarios related to system design. This data collection process was conducted with the needed consent, confidentiality and clearance as part of an ongoing project focused on including IS students at the participating university in the co-design of their IS curriculum. In total the participating university hosts a graduate-level IS student cohort of around 100 students every year. The data used for this study included responses from 34 students from a single cohort, all of whom had selected system design and data analytics as elective courses, and had demonstrated competence in these areas. This purposive sampling ensured that students involved in the discussion of how to embed ethics into their curriculum had a foundational understanding of the ethical issues surrounding data management and system design.

The article is presented as a teacher-to-teacher narrative format combining the insights from student perceptions with the authors' lecturing experience to guide both (a) content and (b) course design. At the time of publication, academic outcomes of the ethics course designed by the authors were not yet available.

The objectives of this paper are thus met, by first discussing content deemed important for inclusion into a graduate-level IS course in ethics, and then wrapping that understanding of "what should go into such a course" with guidelines on how such courses may be built and implemented.

3 Ethics in the IS classroom

In this section, the authors present key concepts and lessons from literature in the form of a teacher-to-teacher narrative on how the task of embedding ethics into the IS classroom may be approached.

The role of IS education is to shape and sustain learners who will be able to effectively design and implement technology in a complex society. The ability of IS to make a significant contribution in solving global challenges, emphasises the importance of ethics in the design and application of technology. This challenge IS students to comprehend the consequences of technology and to be able to make ethical decisions about technology design and use [7].

Topi [3] argues that educators need to develop competent graduates that understand the consequences of ubiquitous technologies on individuals and society in order to avoid harmful consequences and strengthen the benefits of IS. In the design of IS curricula, Topi [3] calls for a comprehensive approach reaching from technical expertise to the implementation of new business models underpinned by a values-based ethical analysis of impact. As argued by Djankov et al., [8] this requires a combination of certain cognitive (critical thinking, problem-solving) and socio-emotional (curiosity, creativity) skills. The authors argue that this scope needs to be increased, because technology design requires a systemic view with an understanding of the moral components of technologies and an inclusive process for stakeholder engagement. Emphasis needs to be placed on the development of aspects such as emotional intelligence, empathy, moral judgement, ethics, personal values and worldviews. As such, Stahl [9] appeals for the development of teaching approaches in IS that allow the development of ethical reflexivity.

3.1 Content

To teach practical ethics, especially the considerations surrounding the responsibility of humans and machines within automated decision-making processes, as part of technical degree courses in IS, students must be given a grounding in how to think ethically. This is a complex task when considering that the majority of IS graduates receive no undergraduate training in ethics and have varying ideas and beliefs about morality.

A detailed list of ethical theories to be considered for inclusion into a graduate-level IS curriculum falls beyond the scope of this article, and acknowledgements are rather made to existing publications of such lists in IS literature. We recognize the work of Burton et al., [10] as a good introduction, grouping ethical theory that relates to IS into three categories: (1) deontology, (2) utilitarianism, and (3) virtue ethics. Literature suggests that an overview of each of these three schools of ethics at the beginning of an IS-focused ethics course is sufficient as a foundation [10] and has the important result of introducing students to the practice of moral and ethical reflection in the IS classroom.

At the level of ethical theory, the list from Burton et al., [10] can be expanded from literature to include a thorough understanding of what (4) moral intuition and (5) explicit morality entail [9]. In summary, ethics theory (content) to include in an IS Ethics course should thus include as a foundation:

Kantian deontology

- Utilitarianism
- Virtue ethics
- Moral intuition
- Explicit morality

Once a foundational understanding of how to think ethically has been set in place, students need to be engaged in reflective practice and debates regarding the ethical issues arising during the design and implementation of new systems. Students may be guided to list issues relating to new technologies – surveillance, privacy, decisions made by machines, the health of users of emerging technology – that they believe to have moral or ethical questions related to them. At this stage, students are still finding their feet as ethical thinkers, and having unmediated debates in the classroom can be counterproductive. A framework is required to guide students and lecturers in productive ethical reflection towards ethically sound IS solutions.

The framework by Leslie [11] brings the ethics surrounding the design and usage of AI systems closer to the realm of system design, translating high-level ethical concerns into a more explicit design framework:

- Fairness (data and design fairness)
- Accountability (responsibility)
- Sustainability (stakeholder assessments)
- Safety (risks of incorrect calculation)
- Transparency (process and outcome)

This framework can be expanded on to include a more extensive list of ethical principles. The list of principles used by the Dutch Knowledge Centre and Data Society [12] was used by the authors in preparing course content on ethical principles: Accountability, Autonomy, Bias, Discrimination, Democracy, Ecology, Ethical "mind-set", Explainability, Fairness, Human dignity, Inclusion, Maturity, Privacy, Responsibility, Responsible use, Robustness, Security, Sustainability, Transparency. We support Hagendorf [13] that ethical thinking needs to transcend the ethical guidelines such as accountability, privacy or fairness to an ethics of care. There needs to be guidance about AI in the context of help, welfare, social responsibility and ecology. Technical solutions cannot be isolated to technical problems and the wider context of relationality and embedded networks need to be exposed to students [13].

3.2 Beyond ethics theory: skills related to reflection and design

With ethics theory and a framework for guiding reflections on ethical principles in the classroom in place, students can be guided to face the challenge at hand in the IS industry: designing AI and related 4IR systems that are ethical by design. The authors found this to be a challenging next step for students, even as they become more adept at ethical reflection. From an initial baseline survey conducted with undergraduate IS (data analytics) students, it was found that, even with a grasp of potential ethical touch points surrounding a system design or analytics project, students failed to think of practical ways to embed ethics into their designs. The problem, as a hypothesis for this article, was that students were not able to break down the task of "embedding ethics into a system design" by allocating responsibility to the: designer, system architect, programmer, data analyst, statistician, data interpreter, or business decision-maker. This skill – the ability to embed knowledge of ethics into designs – comes with practice, and such practice/exercise can only be gained in the classroom from a very hands-on, team-based, scenario-based academic work. The authors probed the concept further in the data collection phase, as presented later in this article.

Literature suggests that students should be exposed to authentic, team-based projects that explore the design of systems that are "ethical-by-design". Monteiro, Leite and Rocha [14], argue in favour of broad and multidisciplinary training that enables an analysis informed by different points of view, using different sets of knowledge via multidisciplinary teams conducting projects with industry. Teams should analyse the impact of technology on complex systems, particularly social, economic and political systems. Likewise, Topi [3] calls for responsible curricula to give students a strong conceptual and practical foundation on ethics that can recognize and address value conflicts and ethical dilemmas to balance human expertise and computing-based automation. Furthermore, Walsham [15, p89] makes an important case for IS students to have a much stronger "ethical agenda of making a better world and a sharper critical agenda towards existing approaches and power structures". As such, literature argues that students need to be exposed to various design considerations adapted from Mulvenna, Boger and Bond [16, p53]

- Design to support the people who will be using the product or service by engendering empathy for users.
- Provide enough information for people to make informed decisions at every stage about whether, when, and how to use the product or service.
- Balance appropriate privacy and security with equitable access by as many systems and people as possible, globally.
- Complement differing needs, abilities, viewpoints and morals.
- Aim for economically, environmentally, and socially sustainable designs.
- Integrate planning for how to handle failure, including transparency and reporting.
- Be realistic about what is possible and needed.

For students to be empowered in critical discourse regarding their views about an ethical agenda, they require ethical reflexivity. Reflection is a process that individuals apply to explore their experiences in order to derive new understanding. We propose the use of Schön's [17] reflective lens that identifies two types of reflection, "reflection-on-action" where students reflect on past activities or actions, and "reflection-in-action" where there is a reflection on actions as they are being performed to question your experience and learn from it. Stahl [5] concurs that ethical reflexivity requires students to grasp their own position and be able to express their views and analyse them from different perspectives. This requires the ability to question one's own insight and the willingness to engage in discourses.

4 Presentation of Data

Throughout the planning and design of a new IS Ethics course (with a focus on Data Analytics and AI) at a South African university, the authors were determined to co-design a course with students that would result in optimal ethical/reflexive growth in students. To achieve this, graduate-level IS students were engaged in several informal group design discussions on ethics, as well as formally surveyed on their understanding of ethics as it related to a range of 4IR technologies. In this section, we present some of the findings from a survey of 34 postgraduate students that informed the design of the IS Ethics course.

4.1 Defining ethics

Students were first asked questions relating to the content that should be included in an IS ethics course. From literature (refer to section 3.1), the authors were interested in their students' understanding of five key ethics theories: Kantian deontology, utilitarianism, virtue ethics, moral intuition, and explicit morality. When asked to rank definitions of the term "ethics", 22 of the 34 students (65%) ranked a definition that translates ethics to moral intuition – a non-reflective "gut feel" of what constitutes good or bad as the best definition. The definition ranked most consistently as second best (29%) was the definition that translated ethics as "virtue ethics" – a theory according to which ethics relates directly to the character of the individual.

The finding that very few students thought of ethics as utilitarianism, deontology, or explicit morals (laws, etc.) sensitised the authors to the reality that students think of ethics – right and wrong, good and bad, truth and lies – as personal; as internal to the individual. As long as ethics is seen as implicit, embedding it into designs is seen as having to translate something personal into a design. Spending time on concepts such as explicit morality, the law, duty, responsibility, utility, and measurable benefits is important. This is an interesting finding as previous studies [10,18] have highlighted that most AI practitioners function within the utilitarian framework. This supports the notion that students need exposure to the major schools of ethical theory.

4.2 Ethics as a skill set that can be learned

Building on the initial realisation that students perceived ethics to be a virtuous character trait, students were asked to rank themselves according to their own perception as having acquired "being ethical" as a character trait. Of the 34 students, 20 ranked themselves as being very proficient at skills relating to moral intuition, and virtues related to having a good character – integrity, trustworthiness, accountability. Students ranked themselves lowest in the two skills categories that related to a (1) utilitarian and (2) deontological philosophical stance. This supported the answers in the first section of the survey (see 4.1). Students saw ethics as personal, and probably because they perceived themselves as having the most experience (proficiency) in solving ethical matters in a very personal way. Skills related to taking an objective, distanced stance to ethical dilemmas and reflecting on the problems using theory was not something students saw as part of their existing skillsets. This finding is expanded upon in section 4.3.

Students were then asked to rank the following three skills in order of importance:

1. To think about (reflect on) potentially harmful consequences of technology
2. To think about (reflect on) the responsibilities and rules governing organisations and systems
3. To think about (reflect on) the personal characteristics required of an IS professional in industry

Students rated option (c) as most important, with option (b) second most important by a single vote.

4.3 The need for ethics when working with future technologies

From literature covered in section 3.2, the authors acknowledged that a clinical knowledge of ethics theory would not complete a student's training towards becoming an ethical IS professional. There are related skills – being able to identify ethical problems in a design, reflect on them, and think of more ethical solutions – that are equally as important as a good foundation in ethics theory. The authors were interested in gauging whether or not students already had some competence in identifying ethical questions inherent to systems and solving them through design.

Students were asked to rate 4IR technologies in terms of ethical risks and ethical sensitivity. Technologies that had to be ranked included AI, ML, bio-electronics, cloud computing, human-machine synthesis, neuro-electronics, quantum computing, robotics, and virtual reality. Students indicated that they perceived these technologies to be ethically risky and sensitive, with 53% of students rating the mentioned technologies as above 8 out of 10 on a Likert scale of ethical risk/sensitivity. AI and robotics scored highest in terms of perceived ethical risk/sensitivity, with student comments indicating a marked distrust of technologies that take decision making control out of the hands of human agents for example:

> *These technologies are ethically sensitive/dangerous because of the speed at which technology is evolving. Many job roles are becoming redundant due to AI and machine learning. The initial purpose of these technologies, in most cases, turn out to be used for something completely different. We are potentially tampering with technology that could out-smart humans and in turn, make us become the experiment.*

Interestingly, several students commented on human emotion as being important during decision-making processes for example:

> *These technologies do not have moral principles embedded in their systems as they are robots, and do not perform and act exactly as humans do. As we are moving towards more technologies performing more and more like humans, it's unlikely that they will perform ethically correct in all instances.*

When the students' distrust of technology is placed within the context of earlier answers relating to ethics being viewed as something personal, something human, a picture starts to develop of the challenges that IS educators face (and will face with increasing frequency) when teaching ethics in the IS classroom. Students don't see systems as ethical, and they don't see ethical principles as "design principles" that can be used to guide system designs. At most, ethics is seen in terms of ethical use – using technology for good, intended purposes. The need for covering ethics in an applied manner – in a way that trains the student in the application of ethics in design, became evident from this section of the survey. One "application skill" that stood out in the literature was the skill of ethical reflection – a concept that was probed further in the next section.

4.4 How to teach ethical reflexivity

Students were asked whether they would describe themselves as “having the skill to objectively discuss an Information Systems problem in a team/group from several different perspectives, including perspectives that differ from your own”. Only 18 students answered yes (53%) and 12 (35%) answered they were not sure of having the skill. Evidently, ethical reflection was not something these students had been exposed to.

Students were asked about the frequency of their involvement in class-based teamwork, project work and case discussions, as well as whether these pedagogical tools would be suitable for teaching ethics. The majority of the students (76%) indicated that they had been well-familiarised with teamwork, project work, and case discussions in the IS classroom, and could see these activities as being good ways of gaining practical experience in applying ethics to IS problems.

When asked to choose teamwork/project themes from a list of ethical principles, privacy was chosen most (14 students), and empathy for end-user consequences – bias, discrimination, fairness, transparency was chosen second most (11 students).

Reflexivity requires introspection and involves a deepened process of reflective action. From the results, it is evident that students have not been exposed the ethical reflexivity and need exposure to ethical theories (deontology, utilitarianism, and virtue ethics) that provide frameworks to reflect on ethical issues and to apply their knowledge to one or more case studies that pose ethical problems [10].

5 Draft design principles for AI ethics course

This article set out to comment in a narrative format on the what (content) and the how (course design) of embedding ethics into graduate-level IS system design courses. A review of the literature, as well as an introductory student survey, were undertaken to develop a set of draft principles for a curriculum that equips IS students to design ethical ML and AI systems. These principles will be tested and refined over three iterations in 2021 and 2022 to further refine the course design. The purpose of the design principles listed below in this article is to serve as a first design iteration – a point of departure – on which other South African IS academics can build. We contribute to the existing knowledge base and provide other practitioners with practical guidelines for implementation for similar interventions in similar settings. The principles are depicted in Table 1.

Table 1. Draft design principles for teaching AI ethics to IS students

Draft Design Principle	**Suggestions for Course Design**	**Suggested Assessment**
Develop a foundation in ethics theory	• Cover Kantian deontology, utilitarianism, virtue ethics, moral intuition, and explicit morality as foundational concepts	Case studies, class debates and presentations

	• Cover the concepts of Floridi et al. [6] to introduce the core opportunities and risks of AI for society and the five ethical principles that should undergird its development and adoption	
Implement a formal process of reflection to encourage reflexivity	• Students reflect learning via a blog, website or vlog. • Students need to write reflective accounts on their understanding of ethical theories. • Include tasks where students need to reflect on the challenges that technological transformations are posing for their future. They need to question their future roles as designers of complex systems for example the power distribution across human, machines, and natural systems. • A further exercise should be a critical reflection on the question of how societies should govern technologies that pose ethical challenges and may have undesirable influences on societal priorities. • Students should also reflect on the personal characteristics required of an IS professional. • Stahl [5] calls for the engagement with moral issues for students to not only understand their position but to be able to formulate and critique their position from a more detached viewpoint to investigate ethical codes of practice and their implications for society.	Lecturer and peers provide formative feedback during the semester, and the formal assessment of all reflections occurs at the end of the semester. Clear guidance in the rubric on expected outcomes. Monitor learnings regarding their reflection-in-action and reflection-on-action [17]
Incorporate a responsibility framework in the systems design process.	• Follow the principles adapted from [13] described in the literature. • Incorporate the principles from [11] as described. • Incorporate design thinking to enable empathetic reasoning during the ideation/investigation and analysis. This will make students more aware of their design imprint on society and the systemic implications thereof.	Formative assessments and feedback throughout the process that emphasise the process and ethical thinking applied in each phase of the systems design process.

	• The concept of responsible innovation should be embedded in the design rollout. Students need to identify ethical problems during design and think of more ethical solutions.	
Implement a service-learning project	• It is recommended to incorporate project-based learning where students work in teams to implement systems in collaborative, community-based projects. • Teams should analyse the impact of technology on complex systems to address 'wicked problems' within society. • Project work that requires them to explore wicked problems will expose them different views and better awareness about complexity and experimentation with multiple design options. • The focus should be on the entire design process and not only the final outcome. Technologies such as AI have the potential to change the world profoundly and irrevocably. Waiting until designs are fully developed to understand and shape their impact is not feasible. • Encourage an experimental approach during the project roll-out. When students struggle and sometimes fail to find a solution, they gain a deeper insight into the problem and its elements.	The outcome of the project is assessed in phases as the roll-out occurs

6 Conclusion

This paper presents findings from a project that explores a curriculum design that embeds ethics in information systems graduate courses. The aim is to foster an ecosystem that reflects ethical codes that govern information system practitioners that are attentive to the implications of ethics for society within the 4th Industrial Revolution. The argument is that technology such as machine learning, artificial intelligence, algorithms, big data, automation and robotics are currently being applied in a wide range of fields and that designers are presented with challenges beyond the current scope of the IS curriculum.

The aim is to develop IS graduates with the capability to understand their role as designers of future technologies, how the technologies they design work and the ethical, legal, and commercial debates these developments are calling into question. Moreover, the implications of 4IR technologies that confront IS students with ethical questions more frequently than in the past. The paper presents findings, within an African context, for design principles that will incorporate ethics within the IS graduate-level AI system design and data analytics curricula in South Africa.

The core design principles to embed are presented as well as aspects to incorporate in the course design. The draft design principles highlight the importance of the development of an ethics theory foundation, teaching approaches that allow the development of ethical reflexivity, incorporation of ethical responsibility in the different phases of the design process and project-based learning steeped in complexity. In conclusion, we propose further iterations of the draft principles to explore the design of a curriculum that equips IS students to design ethical ML and AI systems.

7 References

1. Price R, Matthews J, Wrigley C. Three Narrative Techniques for Engagement and Action in Design-Led Innovation. *She Ji The Journal of Design, Economics, and Innovation*. 2018;4(2):186-201. doi:10.1016/j.sheji.2018.04.001
2. Brynjolfsson E, Mitchell T, Rock D. What Can Machines Learn and What Does It Mean for Occupations and the Economy? *AEA Papers and Proceedings*. 2018:43-47. doi:10.1257/pandp.20181019
3. Topi H. Invited paper - EDSIGCON 2017 Keynote Reflections on the Current State and Future of Information Systems Education. *Journal of Information Systems Education*. 2019;30(1):1-9.
4. Van den Hoven J. Ethics for the Digital Age: Where are the Moral Specs? In: Werthner H, van Harmelen F, eds. *Proceedings of the 11th European Computer Science Summit: Informatics in the Future*. Vienna: Springer; 2017:65-76. doi:10.1007/978-3-319-55735-9
5. Stahl BC. Teaching ethical reflexivity in information systems: How to equip students to deal with moral and ethical issues of emerging information and communication technologies. *Journal of Information Systems Education*. 2019;22(3):253-261. http://hdl.handle.net/2086/6000.
6. Floridi L, Cowls J, Beltrametti M, et al. AI4People—An Ethical Framework for a Good AI Society: Opportunities, Risks, Principles, and Recommendations. *Minds and Machines*. 2018;28(4):689-707. doi:10.1007/s11023-018-9482-5
7. Leidig P, Salmela H, Andrson G, et al. Competency Model for Undergraduate Degree Programs in Information Systems.; 2020.
8. Djankov S, Saliola F, Chen R, Connon D, Cusolito A. *The Changing Nature of Work*.; 2019. doi:10.1016/B978-0-12-641003-7.00003-9
9. Stahl BC. E-teaching - the Economic Threat to the Ethical Legitimacy of Education? *Journal of Information Systems Education*. 2020;15(2):6.
10. Burton E, Goldsmith J, Koenig S, Kuipers B, Mattei N, Walsh T. Ethical considerations in artificial intelligence courses. *AI Magazine*. 2017;38(2):22-34. doi:10.1609/aimag.v38i2.2731
11. Leslie, D. (2019). Understanding artificial intelligence ethics and safety: A guide for the responsible design and implementation of AI systems in the public sector. *The Alan Turing Institute.* https://doi.org/10.5281/zenodo.3240529
12. Knowledge Centre Data and Society. https://data-en-maatschappij.ai/en/tags/rapport-tools-voor-ethiek. Published 2020. Accessed October 10, 2020.
13. Hagendorff T. The Ethics of AI Ethics : An Evaluation of Guidelines. *Minds and Machines*. 2020;30(1):99-120. doi:10.1007/s11023-020-09517-8
14. Monteiro F, Leite C, Rocha C. Ethical education as a pillar of the future role of higher education: Analysing its presence in the curricula of engineering courses. *Futures*. 2018;(February):0-1. doi:10.1016/j.futures.2018.02.004
15. Walsham G. Are we making a better world with ICTs? Reflections on a future agenda for the IS field. *Journal of Information Technology*. 2012;27(2):87-93. doi:10.1057/jit.2012.4
16. Mulvenna M, Boger J, Bond R. Ethical by Design: A Manifesto. In: *European Conference on Cognitive Ergonomics 2017 (ECCE 2017)*. Umeå, Sweden; 2017:51-54. doi:10.1145/3121283.3121300
17. Schön D. *Educating the Reflective Practitioner*. San Francisco, CA: Jossey-Bass; 1987.
18. Goldsmith J, Burton E. Why Teaching Ethics to AI Practitioners Is Important. *ACM SIGCAS Computers and Society*. 2017:110-114.

A Framework for Undergraduate Data Collection Strategies for Student Support Recommendation Systems in Higher Education

Herkulaas MvE Combrink[1][0000−0001−7741−3418], Vukosi Marivate[1][0000−0002−6731−6267], and Benjamin Rosman[2][0000−0002−0284−4114]

1 University of Pretoria, Pretoria, RSA CombrinkHM@ufs.ac.za
vukosi.marivate@cs.up.ac.za
2 University of the Witwatersrand, Johannesburg, RSA
benjros@gmail.com

Abstract. Understanding which student support strategies mitigate dropout and improve student retention is an important part of modern higher educational research. One of the largest challenges institutions of higher learning currently face is the scalability of student support. Part of this is due to the shortage of staff addressing the needs of students, and the subsequent referral pathways associated to provide timeous student support strategies. This is further complicated by the difficulty of these referrals, especially as students are often faced with a combination of administrative, academic, social, and socio-economic challenges. A possible solution to this problem can be a combination of student outcome predictions and applying algorithmic recommender systems within the context of higher education. While much effort and detail has gone into the expansion of explaining algorithmic decision making in this context, there is still a need to develop data collection strategies Therefore, the purpose of this paper is to outline a data collection framework specific to recommender systems within this context in order to reduce collection biases, understand student characteristics, and find an ideal way to infer optimal influences on the student journey. If confirmation biases, challenges in data sparsity and the type of information to collect from students are not addressed, it will have detrimental effects on attempts to assess and evaluate the effects of these systems within higher education.

Keywords: Data Collection Framework · Higher Education · Recommender Systems.

1 Introduction

Improving student retention and reducing student dropout is a major part of 21st century higher education research [8, 38]. In addition to this, understanding minority groups, creating equitable education strategies and reforming curricula to be inclusive in nature form part of the basis of decoloniality seen within developing countries [33, 37, 41]. One of the primary challenges institutions of higher learning are currently facing is the scalability of the staff complement to address the needs of the student cohort [10, 18, 42]. This is further complicated by the complexity of referrals especially from an academic advising perspective as students are often faced with a combination of challenges that influence their academic journey [19, 20]. Furthermore, the challenges students face are larger than previously thought because it includes psychosocial and socio-economic support, in addition to the academic support required [12, 40]. To attempt a potential scalable strategy to address this, the use of digital technologies may be implemented to reach the student cohort. A possible solution to this problem can be a combination of student outcome predictions and applying recommender systems within the context of higher education because the problem inadvertently presents itself to the aforementioned computational tools [23]. Student outcome predictions refer to the academic outcome of students within this context, and the accompanying referral, which if implemented early enough can better support the student. Within the context of the possible solutions to explore in an attempt to address a computational tool or set of tools within higher education, collaborative filtering, demographic based filtering or utility based recommender systems may be applied to this context [34].

Therefore, the purpose of this paper is to outline a data collection framework specific to recommender systems within this context. In order to reduce collection biases, understand student characteristics, and find an ideal way to collect information that can infer optimal influences on the student journey, critical consideration should be given to the challenges and references from previous work in this field.

2 Background

According to Gadinger (2014), immense pressure is placed upon institutions to produce more graduates, and studies in higher education critically engaging with these issues may contribute toward student throughput rates. In order to understand the needs of students, the types of interventions to implement and which student support strategies have been the most successful, student success, student engagement, academic advising, student transition and capabilities of students have to be taken into consideration [38].

One potential strategy that may be incorporated within learning analytics in higher education to computationally assist with the academic advising space, relates to solutions within data science, specifically recommender systems. Recommender systems are platforms that provide a proposition or commendation to users around a set of items. These systems make use of information filtering as the primary methodology to exclude redundant criteria around an item, and include items that are similar to either the user's likes, dislikes, or interests. The application of these systems are widely used in marketing, online shopping and entertainment. Within the education domain, recommender systems provide a feasible solution to the problem of streamlining a few recommendations from multiple referral pathways for a single user [43]. This is especially significant within a context of universities where a multitude of data are generated and stored about the student, in a domain where the recommendation that is made needs to be personalized to the user. Recommender systems function with the premise that data and context are given within a system around two entities, the user (which in this context is the student) and the item (which refers to the intervention presented to the user) [31].

How the information is processed within the algorithm depends on how the input data are filtered. Several different kinds of algorithms exist within this context, including filtering on the basis of: ratings (a user rating about a specific item), demography (race, gender, age, etc.), content data (textual analysis of items rated by the user or multiple users), or item-based collaborative filtering [23].

In item-based collaborative filtering, the objective is to observe a collection of items, denoted by , that the active user, s(u,a) has rated. The items and the ratings are then computed to how similar they are to the target item ij which is then selected from the k most similar items i1,i2,. . . ,ik, based on their corresponding similarities si1,si2,. . . ,sik. Collaborative filtering differs from prediction functions in that prediction functions are expressed numerically (r(a,j)), and are concerned with finding the anticipated opinion of a user (ua) for a specific item (ij) in a process referred to as individual scoring. The output of a successful recommender system is either in the form of a prediction or recommendation. This approach (prediction) in the context of users who are students, however, is counter intuitive within the context of this problem because the scoring is dependent on the user in an environment where the recommendations could be social, socio-economic or psychosocial in nature. Another reason why the prediction approach is counter intuitive is because it takes the mean value between users into consideration without including all of the factors that affect student success, which in turn might not calculate the actual differences between students only similarities based on a few factors. An example of this similarity measure is Pearson correlation-based [28]. This infers that there are several different types of approaches to take when both designing and recommender systems, with ontology based systems as a reference point to include multiple viewpoints within the algorithm design process. Ontology in this context can be defined as a formal, explicit specification of a shared conceptualization [13].

By this definition, an ontology approach by virtue is more inclusive of different ideas because it is fundamentally integrated in the algorithm design process. As promising as ontology-based recommender systems may be, challenges related to biases within their approach have proactively been discussed within the context of human centered approaches [13]. Recommender systems in education have been extensively studied, especially from the application of learning styles as a user rating to an educational intervention [14, 22, 27, 36]. Several questionnaires have been developed for educational recommender systems using learning styles as item based ratings and user rating identifiers, with varying results [9]. The varying results experienced by multiple studies are related to the concept that learning styles in the education do-main have been debunked [2, 21, 26, 29, 30, 39]. However, despite the debunking of learning styles as a school of thought, there is still implementation of the use of learning styles in the field of education based recommender systems[11, 15].

This raises a concern in this field whereby the dispute increases questionable concerns if the underlying data collection instruments are potentially fundamentally flawed in their school of thought. This further elevates disquiets in terms of implementing systems intended to assist the student population, if the required data collection instruments for these systems have not been adapted for this context. Even more so, this is particularly important when considering that learning path generation and subsequent evaluation strategies become contextually difficult if the required instruments do not meet the appropriate metric to be applied for educational recommender systems. Data collection biases are prominent features if the instruments themselves are not applicable for this context. According to Pohl (2004), several biases exist within the context of designing a human centered approach, including confirmation biases, selection biases, implicit biases and reporting biases, just on the virtue of human involvement. In addition to addressing the challenges of bias, fairness and equitability within the context of these technologies need to be considered [5].

The “fairness” component refers to inclusive educational practices that are driven by new fundamental systems in the education domain that focus on including the entire cohort, instead of a specific group within its practices across all pedagogical practices. The “equity” element refers to personalizing elements of the education system to each individual student so that the outcome, not the journey, may be standardized across different students. Therefore, this study focuses on creating a framework for the education domain specifically related to data collection strategies in reducing these challenges.

3 Methodology

In order to understand the different data collection stratagems that are applied to recommender systems in the context of higher education, a desk research methodology was employed. The methodology was chosen to draw strengths from previous works in education, eLearning and recommender systems within this domain. Desk research refers to secondary research conducted on the findings of prior research. The areas of interest in this study are specific to recommender systems and include the data collection strategies employed by previous studies and industry practices. As far possible, data collection strategies specific for education were used, but there were studies that were included in the desk research to strengthen the argument. Furthermore, different recommender data collection strategies are included so as to create a consolidated data collection framework. A theoretical underpinning is required to frame data collection plans to mitigate bias and introduce data collection approaches within this context. In total, three distinct criteria of user item scenarios were explored which relate to data collection considerations, data collection features types and domain specific context Table 1

Table 1. Classification and description for desktop investigation.

Classification	Description
Data collection considerations	Outlining data collection strategies, user specific data collection contexts
Data collection features	Outlining considerations to take into account related to features within the data types collected
Data collection domain specific contexts	Outlining information related to the specific contexts prevalent within higher education

For simplification of the information, the user remains the student in all instances whereas the item differs in terms of the output, information about the user or information related to the specific task performed. Therefore, the recommender should factor in that user preferences might be the result of retrospective evidence about the user as an entity, rather than what the user (in this case student) thinks they require to be successful within university at that specific moment. This methodological concept is supported within higher education, and is even more pronounced when factoring in student transition and the challenges that may arise within society [16].

4 Results and Discussion

Most traditional recommender systems use only one type of recommendation, such as a specific item rating from a set of users. However, as pertained from previous studies, for a recommendation in the learning and education space to be effective, the data collection strategies should be dynamic and able to be evaluated using different types of systems [3, 4, 35, 44]. The dynamic nature of the data collection strategies refers to being able to collect information across various areas of the user experience. Additionally, different types of systems include collecting information from students' pre, during, and post an education activity. Dynamic data collection strategies for educational purposes should include a variety of simple and complex data, allow for dynamic data changes and add not only to the user and item data, but also the domain knowledge (Table 2). The addition of domain knowledge adds to the concept of fairness and equity in the education space [5]. In addition to this, data collection should include the ability to retrospectively study the information that was gathered.

Information collected in the education space related to the specific education recommender problem should be a combination of static and dynamic data that can be used on a variety of scenarios (Table 2). This implies that any recommender data collection strategy should include a variety of dynamic approaches that are inclusive of understanding the education as well as the recommender problem. This means that university leaving students should complete some type of retrospective evaluation survey on the basis of their student journey. This also applies to students reflecting at the end of each semester or module (Fig. 1).

Table 2. Classification and description for desktop investigation.

Data collection consideration	**Recommendations/Areas for future work**	**Sources**
Collect simple and complex data and profile capabilities	Develop comprehensive data structures, include experimental validation processes, select most suitable rating estimation function	[3]
Linguistic pre-processing	Retrospective validation of first study (in their context) needs to be conducted	[44]
Allow users to opt in or out of data collection strategies	Current applications should include multidimensional criteria and more than just the user and item	[35]
Extend user profile, extend item profile, add context and domain knowledge	Intelligent recommender systems using multi context models must be tested on several different types of problems and incorporate different evaluation techniques	[4]

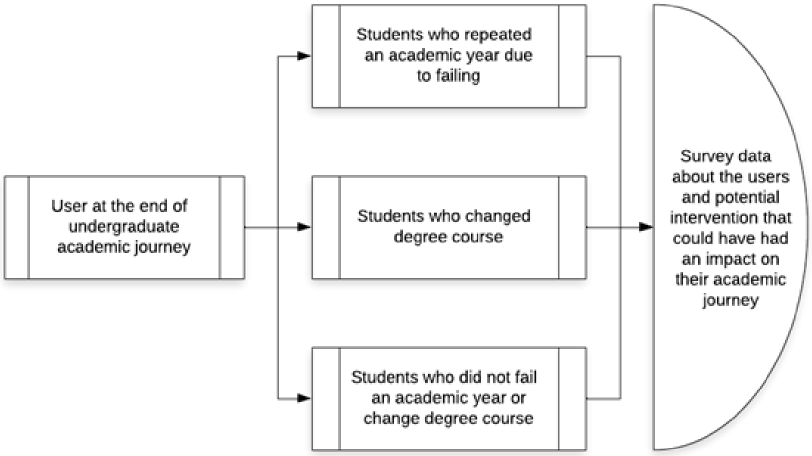

User at the end of undergraduate academic journey
Students who repeated an academic year due to failing
Students who changed degree course
Students who did not fail an academic year or change degree course
Survey data about the users and potential intervention that could have had an impact on their academic journey

Fig. 1. Data academic outcome metric.

However, it must be noted that the inclusion of retrospective evaluation is currently best positioned at the end of the academic journey, rather than at the end of each semester. Furthermore, users should be grouped into three categories, related to major changes in their student journey. These categories relate to students who repeated an academic year due to failing, students who changed degree course, and students who started and finished their initial degree without failing an academic year.

Research into the area of establishing fair algorithmic use so that users can under-stand the biases within recommendations in education has been explored (Table 3). According to Abdollahi and Nasraoui, (2018), “fair” machine learning models are inherently biased on the premise of the algorithm design, depending on the task. Moreover, the data and information that the algorithms are primed from in the con-text of the recommender problem are just as important as addressing bias filtering methods within the algorithm itself [1]. Ontology approaches have been explored in the education, eLearning, and recommender space, with an emphasis on various parts of the systems design, including an importance of data collection [31]. In 2014, the Open University published a "Policy on ethical use of student data for learning analytics" delimiting the nature and scope of data collected, emphasizing an explicit specification on the data that will not be collected and used for learning analytics [37]. This means that the ethics surrounding static data about the user (e.g. demography), dynamic data about the user (e.g. a change in interests), static and dynamic data about the domain (e.g. changes in institutional policy, faculty structures, etc.), and user rating and retrospective evaluation strategies about their own student path-way needs to be established. This is important because in general, evaluations within this specific domain are challenging unless a combination of user-item-domain data that is both static and dynamic is collected (Fig. 2). This further implies that institutional context plays a vital part in this journey.

Table 3. Data collection features, recommendations and sources.

Data collection feature	**Recommendations/Areas for future work**	**Sources**
Data collection adds to the output of recommender systems in education	Ontology-based recommender systems require data collection strategies	[31]
Data collection frameworks are required in education	Policies should be implemented so that data collection strategies at a user level may be used in an ethical manner	[37]
The collection and dissemination strategies were interrogated	Include a user specific explanation to the users, incorporate multidimensional data collection strategies	[1]
A combination of static and dynamic data	Predict user state based on smartphone data and how to convey privacy measures implement-ed to user	[6]
Applied to a static dataset	Design new metrics incorporating additional information related to user scores	[7]
Additional context around the data is required	Where do explanations about variable contribute toward the recommendation	[24]
Linguistic preprocessing	Retrospective validation of first study (in their context) needs to be conducted	[44]

Scholarly engagement within the space of learning analytics may be limited, highlighting unique opportunities to engage in and understand students and the high-er education space better (Table 4). An attempt to understand the challenges students face, identifying contextual differences between students and institutions is on the Higher Education agenda, and grounded in erudite work within these fields. According to Kuh (2008), certain practices promote student success at pivotal moments within the curriculum and undergraduate experience. These interventions are related to high impact practices, but do not extend to social, psycho-social and socioeconomic support. With modernization and an increase in access to technologies popular within ma-chine learning applications, a possible solution to this problem may be rooted in data science. However, within this context several fallacies have been pointed out related to learning analytics within the student context [32].

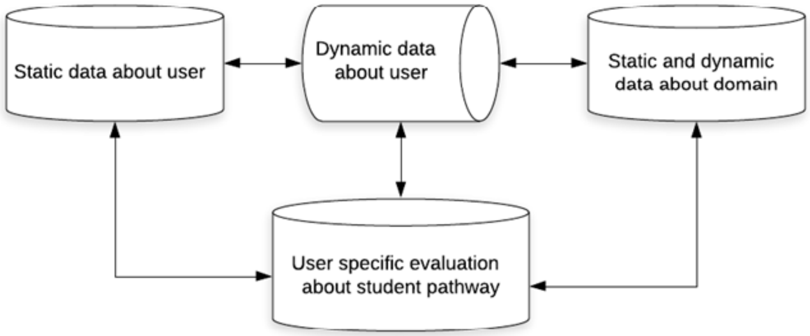

Static data about user
Dynamic data about user
Static and dynamic data about domain
User specific evaluation about student pathway

Fig. 2. Data type differentiation.

These fallacies relate to isolated study design, circular reasoning with no real progress in addressing the problem and a lack of objectivity about the subject matter prior to implementation [32]. Based on their findings, and those before them in this domain, groundwork has been laid to address some of the fundamental programmatic biases that may arise within the understanding of the data that are collected in this setting [17]. To create an ethical and sound basis from which research may be conducted within the education space, data collection strategies about the domain contextualization is required (Table 4).

Within the education domain, student data collection can be grouped into first-year, intermediate, and final-year students. In order for a user to provide an accurate account of their journey, they need to be able to provide contextual information related to their perspective on academe, their field of study, and their university journey. Moreover, there are two streams of information required for each of the three areas of data collection. This relates to the hypothetical and actual account of the user's academic journey. The hypothetical account of the user journey is a retrospective reflective exercise in which the user rates a potential set of interventions that, in their opinion, could have assisted their academic journey. This refers to the actual account of their academic journey, including any academic interventions that occurred (Fig. 3).

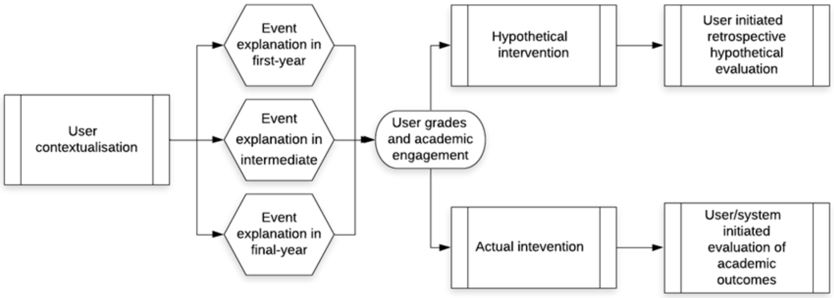

User contextualisation
Event explanation in first-year
Event explanation in intermediate
Event explanation in final-year
User grades and academic engagement
Hypothetical intervention
User initiated retrospective hypothetical evaluation
Actual intevention
User/system initiated evaluation of academic outcomes

Fig. 3. User academic journey contextualization.

In order to incorporate the recommendations proposed by the various authors that were identified in this study, we propose a detailed data collection framework (Fig. 4). This framework represents a consolidation of the groundwork set by previous studies, and consists of three sections related to data collection strategies, data collection variable types and domain specific context. The first succession and strategy within the framework is that data collection strategies must occur retrospectively about the student journey, as well as while the student journey is taking place. With the intention of creating a data collection strategy related to recommender systems in this domain, both the user pathway and the subsequent outcome is required. This is important for testing and evaluation strategies.

It must be emphasized that although data collection from the user is important, institutional data is also required in order to add to this framework to create a comprehensive dataset for the use of studying recommender systems in the higher education domain. This means that the data collection strategy should include a multidimensional data collection strategy from various areas of the higher education system from and by the students. A few potential instruments have been designed that are currently used in the learning analytics space that may be applied to this framework. This includes studies in student engagement, high impact practices and capabilities [25, 38].

Furthermore, routinely collected institutional data should incorporated within these datasets in order to test various recommender system techniques within this domain. Lastly, all of the aforementioned conceptualizations need to be incorporated under the veil of ethics and in the context of studying human subjects [32].

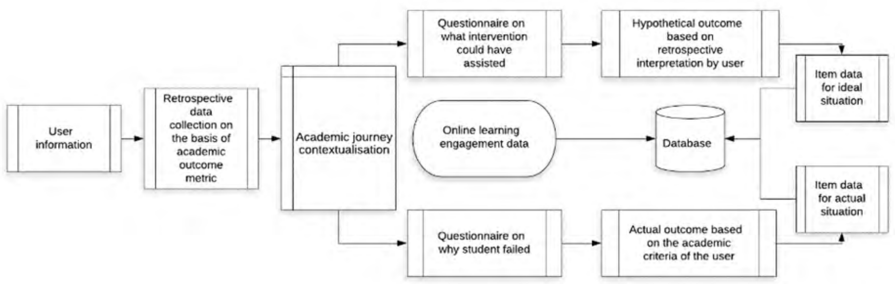

User information
Retrospective data collection on the basis of academic outcome metric
Academic journey contextualisation
Questionnaire on what intervention could have assisted
Hypothetical outcome based on retrospective interpretation by user
Online learning engagement data
Database
Item data for ideal situation
Item data for actual situation
Questionnaire on why student failed
Actual outcome based on the academic criteria of the user

Fig. 4. A framework for undergraduate data collection strategies for recommendation systems.

In addition to this, the proposed framework requires a representation of the student population and student demography profile, implying that a further investigation is required in order to incorporate fairness in sampling [1].

5 Conclusion

If confirmation biases, challenges in data sparsity, and the type of information which to collect from students are not addressed, it will have detrimental effects on attempts to assess and evaluate the effects of recommender systems within South African higher education. If these data collection strategies are not addressed, biases leading to ethical fallacies within this type of research will persist. This study and the purpose of this paper was to outline a data collection framework specific to recommender systems within this context. In South African higher education, the risk of demography based biases may be systemically created, and a subsequent frame-work needed to be created to address this shortcoming. This justification is further emphasized by reducing collection biases and finding optimal ways to assemble information that can infer ideal impacts in the student journey, while not excluding or marginalizing the users. If confirmation biases, challenges in data sparsity, and the type of information to collect from students are not addressed, it will have detrimental effects to institutions, their respective students and society if recommender systems are implemented within this context without the required scholarly engagement. Edizel et al., (2020) justified the inherent biases that exist within recommend-er systems when implemented within society, fortifying societal and systemic biases in a feedback continuum if not addressed. Edizel et al., (2020) further empirically proved that these biases reinforce stereotypes within ethnicity and potentially other societal labels that may be systemically formed. This implies that these processes are complex in nature and require the appropriate engagement in design, evaluation, and implementation in order to mitigate biases within these strategies. Lastly, transparency in fair machine learning occurs at the prediction step of the recommender problem which implies that transparency in system generated feedback needs to be communicated to the academic advisors on the basis of the algorithm design. Within this work, we reviewed a comprehensive evaluation metric for recommender systems data collection strategies within the education domain, and justified the importance of these strategies in terms of evaluation metrics.

6 Implications and Future Research

This research outlined a fundamental step that is required in order to establish the groundwork required for data collection strategies in studying recommender systems within higher education. Future research in this area is required in the form of creat-ing comprehensive and robust data collection instruments as well as establish to what extent the data should be de-identified and stored. Once these instruments are in place, subsequent evaluations of these are required in order to establish which filtering method and systems may be the best fit for the recommender problem in this domain. This also means that the data collection strategies need to include in-formation about the student prior, during, and post their education journey.

References

1. Abdollahi, B. and Nasraoui, O., 2018. Transparency In Fair Machine Learning: The Case Of Explainable Recommender Systems. In: Human And Machine Learning . Cham: Springer, Pp. 21-35.
2. Adey, P. and Dillon, J., 2012. Bad Education: Debunking Myths In Education. 1 Ed. Maidenhead. UK: Mcgraw-Hill Education, Open University Press.
3. Adomavicius, G. and Tuzhilin, A., 2001. Proceedings Of The International Joint Conference On Artificial Intelligence (IJCAI-01). Seattle, Washington, Workshop On Intelligent Techniques For Web Personalization (ITWP2001).
4. Aguilar, J., Valdiviezo-Díaz, P. and Riofrio, G., 2017. A General Framework For Intelligent Recommender Systems. Applied Computing And Informatics, 13(2), Pp. 147-160.
5. Ainscow, M., Dyson, A., Goldrick, S. and West, M., 2013. Developing Equitable Education Systems. 1 Ed. New York, USA: Routledge.
6. Beierle, F. Et Al., 2018. Context Data Categories And Privacy Model For Mobile Data Collection Apps. Procedia Computer Science, 134(1), Pp. 18-25
7. Bobadilla, J., Serradilla, F. and Hernando, A., 2009. Collaborative Filtering Adapted To Recommender Systems Of E-Learning. Knowledge-Based Systems, 22(4), Pp. 261-265.
8. Bond, M. Et Al., 2020. Mapping Research In Student Engagement And Educational Technology In Higher Education: A Systematic Evidence Map. International Journal Of Educational Technology In Higher Education, 17(2), Pp. 1 - 30.
9. Carver, C., Howard, R. and Lane, W., 1999. Addressing Different Learning Styles Through. IEEE Transactions On Education, 42(1), Pp. 33-38.
10. Chasteen, S., Perkins, K., Code, W. and Wieman, C., 2016. The Science Education Initiative: An Experiment In Scaling Up Educational Improvements In A Research University.. Transforming Institutions: Undergraduate STEM Education For The 21st Century, Pp. 120-125.

11. DHET, 2020. Department Of Higher Education And Training. [Online] Available At: Https://Www.Dhet.Gov.Za/ [Accessed 2 September 2020].
12. Abdollahi, B. and Nasraoui, O., 2018. Transparency In Fair Machine Learning: The Case Of Explainable Recommender Systems. In: Human And Machine Learning . Cham: Springer, Pp. 21-35.
13. Ding, Y. and Foo, S., 2002. Ontology Research And Development. Part 2-A Review Of Ontology Mapping And Evolving. Journal Of Information Science, 28(5), Pp. 375-388.
14. Dorça, F. Et Al., 2016. An Automatic And Dynamic Approach For Personalized Recommendation Of Learning Objects Considering Students Learning Styles: An Experimental Analysis. Informatics In Education, Volume 15, Pp. 45-62.
15. Dwivedi, P. and Bharadwaj, K., 2013. Effective Trust-Aware E-Learning Recommender System Based On Learning Styles And Knowledge Levels. Journal Of Educational Technology and Society, 16(4), Pp. 201-216.
16. Edizel, B. Et Al., 2020. Fairecsys: Mitigating Algorithmic Bias In Recommender Systems. International Journal Of Data Science And Analytics, 2(9), Pp. 197-213.
17. El Alfy, S., Gómez, J. and Dani, A., 2019. Exploring The Benefits And Challenges Of Learning Analytics In Higher Education Institutions: A Systematic Literature Review. Information Discovery And Delivery, 47(1), Pp. 23-34.
18. Gadinger, B., 2014. The Influence Of Compulsory Class Attendance On Module Success Rates: The University Of The Free State Case, Bloemfontein: University Of The Free State, Doctoral Dissertation.
19. He, Y. and Hutson, B., 2016. Appreciative Assessment In Academic Advising. The Review Of Higher Education, 2(39), Pp. 213-240.
20. Jaggars, S. and Karp, M., 2016. Transforming The Community College Student Experience Through Comprehensive, Technology-Mediated Advising. New Directions For Community Colleges, Volume 176, Pp. 53-62.
21. Jensen, E., 2000. Brain-Based Learning: A Reality Check. Educational Leadership, 57(7), Pp. 76-80.
22. Klašnja-Milićević, A., Vesin, B., Ivanović, M. and Budimac, Z., 2011. E-Learning Personalization Based On Hybrid Recommendation Strategy And Learning Style Identification. Computers and Education, 56(3), Pp. 885-899.
23. Konstan, J. and Riedl, J., 2012. Recommender Systems: From Algorithms To User Experience. User Modeling And User-Adapted Interact, 1(22), Pp. 101-123.
24. Kouki, P. Et Al., 2019. Personalized Explanations For Hybrid Recommender Systems. Marina Del Rey, Proceedings Of The 24th International Conference On Intelligent User Interfaces.
25. Kuh, G., 2008. Excerpt From High-Impact Educational Practices: What They Are, Who Has Access To Them, And Why They Matter. Association Of American Colleges And Universities, 14(3), Pp. 28 - 29.
26. La Lopa, J. and Wray, M., 2015. Debunking The Matching Hypothesis Of Learning Style Theorists In Hospitality Education. Journal Of Hospitality and Tourism Education, 27(3), Pp. 120-128.
27. Liegle, J. and Janicki, T., 2006. The Effect Of Learning Styles On The Navigation Needs Of Web-Based Learners. Computers In Human Behavior, 22(5), Pp. 885-898.
28. Mooney, R. and Roy, L., 2000. Content-Based Book Recommending Using Learning For Text Categorization.. San Antonio, TX, USA, Proceedings Of The Fifth ACM Conference On Digital Libraries.
29. Newton, P., 2015. The Learning Styles Myth Is Thriving In Higher Education. Frontiers In Psychology, Volume 6, P. 1908.

30. Newton, P. and Miah, M., 2017. Evidence-Based Higher Education–Is The Learning Styles 'Myth'important?. Frontiers In Psychology, 8(1), P. 444.
31. Obeid, C., Lahoud, I., El Khoury, H. and Champin, P., 2018. Ontology-Based Recommender System In Higher Education. S.L., Companion Proceedings Of The The Web Conference .
32. Pohl, R., 2004. Cognitive Illusions: A Handbook On Fallacies And Biases In Thinking, Judgement And Memory. Ed.: Psychology Press.
33. Sayed, Y., Motala, S. and Hoffman, N., 2018. Decolonising Initial Teacher Education In South African Universities: More Than An Event.. Journal Of Education (University Of Kwazulu-Natal), Pp. 59-91.
34. Schafer, J., Frankowski, D., Herlocker, J. and Sen, S., 2007. Collaborative Filtering Recommender Systems. In: The Adaptive Web . Heidenberg: Springer, Pp. 291-324.
35. Shaharin, R., 2017. Private Recommendation: Extending Capabilities Of Privacy-Protection Recommender Systems. IJCSIS, 15(4), Pp. 267-274.
36. Shen, L. and Shen, R., 2005. Ontology-Based Learning Content Recommendation. International Journal Of Continuing Engineering Education And Life Long Learning, 15(3-6), Pp. 308-317.
37. Slade, S. and Boroowa, A., 2014. Policy On Ethical Use Of Student Data For Learning Analytics, Milton Keynes: Open University UK.
38. Strydom, F., Kuh, G. and Loots, S., 2017. Engaging Students: Using Evidence To Promote Student Success. 1st Ed. Bloemfontein: AFRICAN SUN Media.
39. Tokuhama-Espinosa, T., 2018. Neuromyths: Debunking False Ideas About The Brain. 1st Ed. New York, USA: WW Norton and Company.
40. Ulriksen, L., Madsen, L. and Holmegaard, H., 2017. The First-Year Experience Of Non-Traditional Students In Danish Science And Engineering University Programmes. European Educational Research Journal, 1(16), Pp. 45-61.
41. Upcraft, M. and Gardner, J., 1989. The Freshman Year Experience. Helping Students Survive And Succeed In College. San Francisco: Jossey-Bass Inc., Publishers.
42. Upcraft, M., Gardner, J. and Barefoot, B., 2005. Challenging And Supporting The First-Year Student: A Handbook For Improving The First Year Of College. 254 Ed. San Francisco: Jossey-Bass..
43. Vozalis, E. and Margaritis, K., 2003. Analysis Of Recommender Systems Algorithms. S.L., The 6th Hellenic European Conference On Computer Mathematics and Its Applications.
44. Wiesner, M. and Pfeifer, D., 2014. Health Recommender Systems: Concepts, Requirements, Technical Basics And Challenges.. International Journal Of Environmental Research And Public Health, 11(3), Pp. 2580-2607.

The Implications of Personalisation Algorithms on Individual Autonomy

Heather Rufus$^{1[0000-0003-2842-3411]}$

University of Witwatersrand, Johannesburg

Abstract. There has been an exponential increase in the amount of information avail-able online that would be impossible to make sense of without some means to organise and filter it. Informational filtering – of which personalisation algorithms form part of – have been developed, by social media and search engine platforms, as tools for this exact purpose. Personalisation tailors the information an individual can see by monitoring and analysing an individual's online activity and data to predict what information will be most relevant for a particular user. However, personalisation faces a number of ethical issues around topics such surveillance. One issue that has been under-explored are the implications of personalisation on individual autonomy. This paper investigates whether the control over the dispersal of information to individuals brought about by personalisation can be seen to infringe on individual autonomy or if it could, in fact, enhance autonomy.

Keywords: Personalisation Algorithms · Autonomy · Transparency.

1 Introduction

Our world has changed. It is becoming almost trite to point out how quickly our technology is advancing now, how many things are different now to how they were 'then' - whenever 'then' refers to for whomever. The changes can be seen everywhere you look. Change is not new to us as a species. Our evolution throughout the ages bears testament to our ability to innovate and adapt. These are not new observations nor are the struggles emerging in our attempts to under-stand and guide the developments coming to shape our world. These struggles have met their same adversaries only now in a new form. The rapidity in our digital technological advancements has seen the rise of a flow of information at such a scale that far exceeds what and how information was available to us before. Personalisation algorithms are an information system, a set of instructions helping to order, organise, and 'deliver' information, available to us but designed in often opaque ways for uncertain purposes. When designed in this way, it is plausible that personalisation algorithms do not allow for access to information in a way necessary for autonomy. It could be argued that if we are to be autonomous, then the information systems available to us to inform our thinking and decisions must be designed in such a way that is transparent and to the user's benefit to safeguard its use to manipulate rather than inform.

However, the power of personalisation may present an opportunity to use its efficiency in such a way that enables autonomy rather than undermines it, as will be argued in this paper.

In order to argue that personalisation algorithms can in fact enable autonomy it is necessary to have an understanding of both personalisation algorithms and autonomy to hand. In section 2, what personalisation algorithms are and why they are used is outlined. In section 3, the general reactionary worries are covered. In section 4, the ad-vantages of using algorithms will be discussed. Following that, in section 5, the potential cost of these benefits will be assessed before introducing how autonomy can be understood, and pertinent, to this analysis in section 6. Sections 7 and 8 will consider the implications of how personalisation algorithms function on autonomy and potential ways its drawbacks may be reversed to enhance rather than impede on autonomy.

2 Personalisation Algorithms

To answer whether or not personalisation infringes on individual autonomy, a critical overview of what these algorithms are and how they work is necessary. There is a huge proportion of the global population that spends a large quantity of their time every day engaging with information online which is, essentially, to make use of algorithms in some manner or form. It is estimated that over sixty percent of the world's population uses the internet [18]. This also means that the quantity of data produced as a by-product of that computing is staggering [15]. Algorithms form the engine of the online platforms that we use daily. These can be broadly categorised as falling under Artificial Intelligence (AI). They run the sites, implement us-er queries and retrieve information. Facebook and Google are two of the most popular sites used in the world and the average individual can speed up to seven hours online every day (Ang, 2020; Salim, 2020). As such, the more popular and prolific the platforms the great-er the level of access to individuals across the world.

Personalisation algorithms function as a form of informational filtering. For the algorithms to function, they draw on an individual's specific user profile. User profiles are determined by an individual's basic information and usage mining in which one's past interests, locations, and click history, are tracked to develop a data set that can be analysed and used to produce results on search engines and social networking sites that will be relevant to the user [5] [9]. Exposure to large amounts of information impedes on the individual's ability to decipher which information is useful or relevant and so as a result leads to information overload or over-stimulation [5]. Given the sheer amount of information available online, some form of informational filtering becomes necessary to access information in such a way that can be meaningful. Meaningful in this sense refers to one's ability to find information without requiring great levels of time or the overcoming of many obstacles. Part of what makes information 'sensible' is the ability to access it without becoming overwhelmed.

Whilst personalisation algorithms form part of the service that draws individuals to using platforms, it is also is part of that which makes the companies running these platforms profitable. The more individuals who use the platforms the more value for both people and company. The network effect is a phenomenon in which a service or good increase in value the more people who use it [2]. This sets up an interesting relationship between an individual and the companies providing these platforms to the public as companies driving the establishment of various platforms online are inclined to optimise the experience individuals have on their platforms to reap the rewards of the network effect. There is an abundance of platforms and so to stand out, to keep people's intention, the platforms are designed to appeal to our most base, impulsive, instincts, to ensure that the way individual's environment is formed makes you want to use these plat-forms frequently. This attention economy characterises the general power dynamic between users of digital technologies and technology companies in which the latter design its offerings to keep individuals engaged to maximise mass attention and, consequently, profit [17].

There is a long history of information filtering filtered through by scores of editors, journalists, community elders. There is a huge pro-portion of the global population that is spending a large quantity of their time every day engaging with information online but these traditional gatekeepers with our traditional institutions keeping them in check have not yet had the time to develop. Policy-making and regulation lag behind the speed at which technologies develop. As such it is important that investigations such as the effects of personalisation algorithms and autonomy are had for, they contribute to the negotiations of how these systems will be allowed to work. One such negotiation with regards to how the systems work pertains to surveillance through personalisation.

3 Personalisation, efficiency at what cost?

The efficiency of accessing quick, relevant, information may seem free but there is a trade. The use of these platforms may seem free but the trade is the access we grant such platforms to enable the possibility of surveillance. There are concerns around the use of data for surveillance by focusing on the intersection of individual and 'the internet', meaning all that can be encompassed within that term, in the form of personalisation algorithms. The cost of the use of 'free' platforms such as Facebook and Google is data. Each time someone uses platforms such as these there is a wealth of information about the individual which can be used for a variety of purposes. One of the main driving motivations for collecting and analysing this kind of data is that it allows for targeting. Or, in other words, surveillance. Surveillance is understood as the systematically observing individuals, by means of the processing and analysis of multiple sources of data pertaining to an individual's online activity [15]. When this is done at a mass level, surveillance becomes dangerous because it gives a few entities the power and information to discriminate, exploit, or manipulate a large number of people [15]. Access to the individual is tantamount for success. It determines which platform thrives and how those platforms are designed.

A tool like personalised targeting (or algorithms) is that targeting will allow for optimal access to information, products and interactions that are relevant to a person. There is a motivation to keep the service as efficient as possible firstly, to keep users using the platform and, thus, retaining the value of the network effect, and secondly, to optimise the profits that can be generated through these services. But the databases and analytic practices that have come to exist cast shadows on such optimism. This motivation to power profit from surveillance is made possible by how individuals across the world have digitised their lives. The success of these companies is largely attributed to their ability to attract and maintain attention (read: data) supply [13]. Personalisation algorithms are part of what compels our attention. After all, we are far more likely to invest our time and energy in things that feel relevant to us. Consequently, this level of access to individuals is cause for concern because it allows for surveillance of individuals, their activities, interests, and relationships, on a previously unheard-of scale.

After all, the development of information is never a neutral process. Information that survives is that which is useful, relevant, and valued, by individuals and society. An individual's use of different online plat-forms can indicate what their interests are. The idea is that having something like a personalisation algorithm reduces the burden of accessing large quantities of information and makes what one finds more likely to be what they were searching for. How information comes to be tailored to the individual is not something that the general public who utilises these services is aware of. Various third par-ties, such as advertising companies and governments, have taken interest in personalisation algorithms because of how its potential to be used to influence individuals through personalised targeting. The result is obscure practices in how targeting is set up, and data is mined as well as the subtle prioritisation by platforms of the content of other companies they have partnered with ([19] [5] [9] [8]. But the evidence suggests that user relevant results are not solely the way personalisation algorithms are designed.

Users are seeing different things depending on their online usage history and some of the information they are seeing are not necessarily the facts of the matter but information about the matter tailored to them in a particular way or for a particular purpose. It may not be drastic: content relevant to what the individual was searching for was shown and directs traffic to a partner's site. This type of content is how a platform like Google to derive some sort of value from showing biased information. However, given that personalisation is part of what connects the entire ecology of plat-forms online in some way or form, it has the potential to influence how most people access information about their social circles, politics and news – to name a few.

4 Implications of Surveillance

It seems then that there are private agendas determining how online platforms function. Bearing such in mind, it becomes unclear whether one is shown information that is valuable in its own right or whether one is slowly being pushed towards one source of information rather than another due to some agenda that the platform might have. This is something we should not lose sight of particularly as we become increasingly dependent on them for information. The opacity of this system and our dependence on it is a threat to individual autonomy for it obstructs access to unbiased information necessary for autonomy to arise. If all the information one is receiving is designed to influence you in some unknown way, one is being manipulated.

The effects personalisation algorithms have on our thinking is already apparent in studies researching the effects of filter bubbles. Where we have been looking at potential modes of manipulation, one case study already exists. Filter bubbles are understood to form as a result of an individual's interests. As such, users are consistently only shown information that it is predicted they would be interested in or would tend to agree with. As a result, individuals are not aware of other events, counterarguments and general information falling outside that bubble. Many users are unaware that their search results or newsfeed are personalised in the first place and, thus, are unaware that the results they are seeing do not object. Personalisation can also mean that certain information is hidden from users as a result [9].

Human beings seeking out like-minded companions or information one would agree with is not a new thing. It is a bias that can be identified in various aspects of social life. It can be amplified through personalisation where one does not even have to seek out agreeable in-formation – it is already available. The friction is removed, and the concern is that people will be less and less likely to seek out counter-arguments. The environment is designed to encourage engagement with information one would agree with. One is nudged towards this kind of information. One could seek to 'pop' one's filter bubble by seeking out counterarguments for oneself – which many individuals do – but more often one uses the information as it is presented to one-self. Either there is a lack of awareness of the filter bubble or lack of motivation to care or change the situation. We seem inclined to accept the deal we have because of the benefits we reap. But the design of this system is hurting us. Perhaps in invisible ways now but we are paying in various ways. Autonomy is something particularly dear to part with and it seems personalisation algorithms are part of the Faustian deal we are trading it in for.

5 Autonomy and Nudges

The societal structures we are immersed in can play a role in limiting the field of an agent's possible action or choice in ways that can infringe on autonomy. Personalisation algorithms organise the environment that informs how people make decisions. [19] draw on ‘nudge’ theory to argue that people’s behav-iour is easily swayed by subtle factors in their environment such as personalisation which may threaten autonomy as it could lend itself to targeted manipulation. However, a nudge is something one can choose to ignore. To opt-out of. As it stands, there is no way to verify whether one is being pushed towards some information rather than others and, if they are just being nudged.

Environmental nudges such as those issued by societal norms and beliefs can influence the kind of action an individual takes [7] [16]. The available array of choices to an individual is in part dependent on their own ability to make decisions as well as partly informed by the environment around them.

One way to understand how autonomy is constituted is to see it differentiated by two orders [7]. Second-order autonomy is understood to refer to an individual’s ability to live her life in a manner of her choosing, her choice informed by her notion of the good, and protected from any coercion or influence that would seek to sway her from choices that would bring this life about for her [7]. First-order autonomy is understood to be held by an individual who can set rules she wishes to live by without compromising on her ability to question, critique, and ultimately reject these rules should it be necessary [7]. To have autonomy in both senses is, thus, the ability for an individual to choose how she wishes to live her life and by which rules protected from coercion and so on. The individual can deliberate on the nature of the rules she endorses which guide how she lives her life without compromise. She can choose alternative rules that are better suited to what she wishes herself and her life to be and this choice is made as a result of her own reasons rather than what someone else would will her reasons or choice to be.

However, second-order autonomy is that which is usually taken up over first-order autonomy. It is in this way provisos are made for those individuals who may decide to autonomously choose for themselves non-autonomous lives. By some accounts, the individual’s autonomy remains intact depending on the conditions that bring about that choice in the individual. That is to say: so long as the individual took up this non-autonomous way of life by their own volition. An example of such an instance would be an individual who chooses to take up a way of life that is heavily regimented with strict limitations on what the individual may or may not do. Ways of life such as those followed by the military or religious institutions are indications of where individuals have, and do, take up a life which impedes their negative liberty as well as first-order autonomy with second-order autonomy.

Second-order autonomy without the first-order autonomy disallows the individual the ability to question the rules of nature, of the life one has and depart from those rules if they are not ones the individual would want to endorse for themselves [7]. According to [7], the tendency to prioritise autonomy of the second-order over that of the first order overlooks how the choices that individuals make may be shaped by the greater social influences or structures within which they are immersed. Autonomy requires an agent who can choose how one is going to act. We can retain the ability to choose autonomous courses of action even when aspects of our environment around us are determined by forces, not of our choosing nor our making. To choose one's way of life requires that one has reasons for why one way of life would be preferable to another way. To act with autonomy is understood as when an agent is understood to act freely when her actions follow from reasons that are her own [11]. An agent's free actions are conditional on the agent's choices. As I have discussed above, an agent's choices are susceptible to societal or environmental influences. An agent has the capacity to navigate those influences and form her reasons so long as those influences are determined in such a way that an agent can choose her act differently.

One can be thought to have control over one's actions when one can deliberate on which reasons one has for action and which of those would be best to act upon. The deliberation process by which an agent comes to evaluate her reasons is informed by her beliefs, reasons, and values. One chooses one's reasons for action in part of the basis of one's values, beliefs and desires. These states are not entirely within the control of the agent to determine. Such states could be biologically or societal informed. However, so long as these states are brought about by the agent and not as a result of manipulation or brainwashing then free action is possible [11]. This is possible because the agent has control over the action they take and to decide upon that action they appropriately reason sensitive [11]. That is to say that the agent can evaluate the factors at work in particular instances and evaluate them to determine what her best reasons for action would be.

For an agent to have autonomy of the will, the agent must form her beliefs in relation to what is true and the evidence proffered for that view. Her desires must be formed in relation to her needs and those needs formed in relation to evidence for the good. But she is not determined by those. The must be in control of the actions that follow even if that means she takes the action of reasoning further. This is possible so long as these states informing an agent's reason for action are not brought about as a result of brainwashing or manipulation [11].

Humans are inherently social creatures. We do not exist within a vacuum. Part of our person-hood is formed in relation to the language, culture and norms of the particular locations in which we occupy. Oft times people will follow the norms of their contexts with second-order autonomy because of the social benefits it brings them. Or perhaps, it would be better to phrase it in this way: sometimes people follow social norms not necessarily because of the social benefits it will bring them but avoid the costs should they not follow those norms. In some of these instances, this involves foregoing first-order autonomy. In our highly digitized world, this could explain why so many individuals have accepted the current data-usage model of online platforms. The model has become normalized and where we initially lacked the insights to challenge its establishment, we now seem too dependent to pose a serious challenge on its flaws.

If we are nudged into use of this model, within which manipulation not nudges can take place, we can make use of this same mechanism to design it to be more beneficial to individuals. We exist surrounded by structures of our species' designs. And no design is neutral. Bearing that in mind, we can claim back our control over the designs that have such significant impacts on our everyday lives. Nudge theory notes that how an environment comes to be organised always influences the nature of people's choices within that environment [16]. The way cities, kitchens, offices and news-feeds are never designed arbitrarily, and it can be the case that how environments such as these are designed unintentionally influence people's choices in ways that may not be in their best interests. Thaler and Sunstein [16] argue that these environments should, and can, be intentionally designed to facilitate better decision making but never coerce individuals into making those decisions. Individuals should always be free to choose and act as they wish, to opt-out of situations not to their liking and to be protected from coercion [16]. This would be in line with the Chamber's notion of having both first and second-order autonomy.

If one is always influenced in one's decision making to some degree, nudge theory would advocate trying to design the environment in which a person makes decisions in such a way that facilitates better decisions and, as such, gently nudges them towards choosing those options. These 'nudges' can be ignored but they serve as suggestions or reminders of one's ability to choose differently. Though nudge theory is positioned as something to enhance one's decision making in one's best interest, it is also possible that such a phenomenon can be used to one's detriment.

6 Application of Personalisation to Surveillance

The same logic of nudge theory could be applied in the general data model in which personalisation algorithms are situated. The environment, however, is not designed to nudge but rather to push. Is this why we have gone along with how personalisation algorithms have been developed? Why such pervasive levels of surveillance have been allowed to accrue? Though we may have autonomously chosen this path, the role social structures play in influencing an individual's action should be guarded against particularly when it has the power to be totalising.

Personalisation poses a threat to autonomy in three ways (though there could potentially be other issues) due to a lack of transparency, control, and choice. The lack of transparency in how the algorithm uses and tailors information to users could result in people being influenced or manipulated in ways that they are not aware of. There is a lack of user control over how or when personalisation works. Moreover, users may lack the choice to agree or opt-out of using the algorithm in the first place. We rely on information to inform our decisions about who we want to be, what we believe and what we do within our everyday life. If one of the predominant sources from which we draw information could potentially manipulate the reality of things, could restrict the field of possible action, our own actions and beliefs could similarly be manipulated.

Whilst people may be autonomously choosing to use online platforms, this preference is in part informed by the fact that it has become the norm. Moreover, there are social costs not just to not 'being online'. In addition to this, there is a greater cost to privacy and, as this account is concerned with, autonomy. The opacity of how these platforms functions give them great power over individuals which may result in manipulation. It is certainly becoming the case that we not have autonomy of either first or second order to question the rules by which we are now coming to be governed, free from influence.

The question then arises: is the threat of personalisation algorithms to autonomy a design flaw or a systematic problem? Can there be autonomy in a world in which we rely on online informational distribution through information filtering systems we have had no hand in determining their uses? Is not the mere reality that information is determined and accessible to us through forces beyond our control a determining factor that would necessarily undermine autonomy?

Now if the factors available for an agent's consideration are being unduly influenced by something like a personalisation algorithm then an agent's ability to form her best reason is being influenced by external, unclear, motivations. I say 'unclear' because the mandate instruction could be profit, political, or some personal bias, but there is no transparency by which to ascertain which.

For an individual to be capable of critically engaging with the norms and structures in one's life and to be able to do so through assessing evidence means one is autonomous. Should these capabilities be obstructed, so too is her ability for autonomy. There are instances where individuals lack autonomy. An agent's free action arises through an agent's considered and valued reasons or choices and had the individual had different reasons it follows that she would have acted differently [11]. However, the individual does have control in choosing to act to find more evidence or take better action. If this pursuit itself only yields other biased evidence (in personalised search results for instance) then the individual's ability to exercise their autonomy and find better evidence is interfered with. One does not have to be in control of what information is available to oneself as it is not possible for one to entirely control what information is available to oneself.

Practically, there is some level of external control over whatever information is available to individuals whether it be for historical or practical reasons like a book not being held by one's local library or a certain narrative of reality being upheld within a particular culture. The important distinction is then that the information available should be as unbiased and fair as possible. It should not be designed to manipulate – or perhaps in its extreme form: brainwash. This is also a contentious statement because, as the discussion on algorithmic bias demonstrated, bias is inevitably human. So a qualifier is then required to differentiate between information that is controlled in the sense that it has been filtered is some sort of way versus information that has been intentionally designed to be biased, to manipulate, to brainwash even. The latter case would undermine the autonomy of an individual's deliberations.

Information that has been filtered to help one find answers to a query does not inhibit the necessary elements required for autonomy *per se*. However, as discussed above, it can also be designed with not that function alone in mind. The feedback loop between individual interests, relevant data to keep one engaged and that in turn yields more data and then more profit is such a design. The way that the internet has evolved has facilitated a model in which individuals are not hugely concerned about what data or activity of theirs is being monitored. It has just become the norm that this is how platforms operate and there is an enormous benefit to be had by using these platforms. This makes individuals more likely to accept a model and overlook for instance problems with mass surveillance or privacy violations. Individuals are not forced to rely on the internet for information – social, political or otherwise. There are alternative ways. However, in a world that is becoming increasingly digitized it is becoming the standard medium or, otherwise put, a social norm. However, though we may autonomously participate in this way of life (with second-order autonomy), there is an informational asymmetry between users and the platforms which may result in our lacking autonomy of the first order.

7 Personalisation, friend or foe?

There are thought experiments in the philosophy of mind debates about different instances where an individual is brainwashed by some other entity to behave a certain way so as to achieve a certain outcome. Sometimes it is Martians, other times some omnipresent being. The point is that what these debates demonstrate is that more often than not such types of brainwashing impede on the autonomy of that agent. This is a lot closer to home now than a thought experiment. This section explores the practical consequences of personalisation algorithms when they run unchecked.

Cambridge Analytica's scandal in the 2016 US election reveals how vulnerable we are to manipulation in ways that transcend the advertising industry's use of personalised advert targeting. We are vulnerable in ways we are otherwise unaware of. So often the speed at which technology advances outmatches our ability as human beings to catch up with its implications on society. As such we are often playing catch up to forces that significantly affect our world. The Cambridge Analytica saga is detailed by the data breach – and use - of 50 million Face-book user data points by the company Cambridge Analytica in the 2016 US election to target key voters with personalised political advertising leading up to the election (Cadwalladr & Graham-Harrison, 2018). Inferences about individual political preferences were made through their user-profiles and advertisements, playing on certain vulnerabilities of these individuals, were targeted to individuals in key voting areas within the USA [6]. The use of this strategy within the election is attributed as being integral in swinging the election in Donald Trump's favour. The power of personalisation to be used to destabilise real-world institutions reiterates how important it is for our information sources to reputable and transparent. This is essential for individuals to be able to use that information to inform autonomous action and choices.

The lack of transparency around how information is personalised poses a threat to autonomy as people may be influenced in ways that they are unaware of. Nor could they be aware of it, given how covert its effects are. The power of targeted advertising already stands as evidence of the efficacy of personalisation as an influence. This may be focused on consumer purchases now but can use the same reach it has to influence us with respects to our knowledge about the world, politics and, perhaps, even morality [19]. But if we do not understand how this technology works, if we cannot hold the platforms accountable, then we have to realise that we are vulnerable to private companies – with their own various agendas, nefarious or otherwise – in how they represent the world around us.

The choices or actions that individuals make should follow from their reasons for action which are – in part – determined either through social structures or information. The lack of control users have over personalisation does, in part, stem to the problem of the lack of transparency. The lack of user control could mean control of the user. If the information dispersed to individuals as a result of personalisation has been determined by interests other than the individual's we may consider it as being biased. That is to say that if the information one is being exposed to is a result of a platform or advertiser pushing certain content to individuals rather than depicting a range of information available, the information can be considered biased in this regard.

We as individuals, as citizens, of the technology era may choose to participate in the use of online platforms like Facebook and Google, but we cannot question, or effect change over the rules of that engagement as first-order autonomy would necessitate. The consequences of this are that we could come to lack second-order autonomy, too. If we have no say over the rules of a way of life significant to many individuals, no control or insight into what determines the in-formation which partly informs how we make decisions and take action in our own lives, then how sure can we be that we are free from coercion or influence in the ways that we come to live our lives? It can be argued that individuals should just forgo these platforms altogether.

There are alternative platforms that are beginning to emerge that have been formed with principles such as transparency and the avoidance of unnecessary data collection at their core. Search engines like DuckDuckGo or a social networking platform like Diasphora. However, the uptake of sites with philosophies like this is slow. This could be attributed to multiple reasons one of which being the network effect. The popularity of 'mainstream' platforms is ironically part of what keeps people there. Another cause could be that for the most part people are unaware of their vulnerability to manipulation, as was the case in Cambridge Analytica, and so see no need to look to alternative platforms for their various needs online. A practical obstacle which may discourage people is the additional burdensome of these plat-forms have by either requiring some technical knowledge such as cod-ing or being less 'efficient' because it lacks the touch personalisation algorithms so many have become accustomed to.

There is a significant part of life that is lived offline. However, the spread of information between countries, communities and people is increasingly occurring in the online space. There is a coterminous relationship between the two spheres. Transparency, control and choice are aspects of personalisation algorithms that are possible to im-prove. It is possible to use the relationship between these two spheres of human life to supplement and enhance one another if done correctly. To have a friend rather than foe.

8 Personalisation towards Autonomy

If we are to ensure that our environments are designed to ensure the possibility of autonomy, then our ability to access information must be a process that is free from manipulation. Human beings are not perfect decision-makers. We are not entirely the rational and self-reflexive subjects that classical Western philosophic traditions would have us be. Sometimes, even when we set out to make decisions based on our best and most well-considered reasons for action, we fail. This section will discuss how we can use personalisation algorithms to help us make guard against the weaknesses that can cause us to fail.

Algorithms prove more consistent in weighing factors necessary for decision-making. They are less prone to human cognitive errors – should the coding be done in a fair and evaluated manner – and can process far greater amounts of information with an accuracy that surpasses human ability. These are a few reasons of why we would want to integrate algorithms into our everyday life. Granted, biases can still be entrenched within algorithms as a result of their human designers. However, the ability of algorithms to consistently make reliable judgements in the face of partial or uncertain information is better than that of human beings who can be easily swayed by ulterior environmental considerations not entirely pertinent to the judgement at hand. Heat and one's mood, for instance, have been shown as relevant factors in how individuals go about making decisions whilst algorithms can be designed to consider only what information is identified as relevant in this particular instance [10]. We can design our tools to aid us where we are flawed, to guard against where we make errors. A personalisation is a tool that we can use in this sense.

The information age is only useful to us, as humans, if we can use the information. At all stages of our evolution, we have found tools to help us. Personalisation has the potential to be such a tool. It is worth noting that even in our development of tools, we often fall prey to cognitive biases. However, it is possible to become aware of those biases, to understand and guide our thinking to reduce our own errors in judgements and decisions [20]. The success of the internet throughout its iterations is also evidence that we are not perfect decision-makers, that we are vulnerable to the exploitation of our shortfalls [17]. Our ability to fail, to re-evaluate the evidence and try – maybe even fail – again are important facets of exercising autonomy. Our inventions are also testimony to that fact. It is in our ability to be imperfect and yet continue learning, re-trying, that we exercise autonomy. It is also the same curiosity and ability for analysis that allows us to come to grips with our weakness-es (and certainly what has allowed the attention economy to unfold with the hegemony that it has). There are studies into the types of cognitive biases [20] [10] that illuminate the inconsistencies in our thought. That allows us to refine our perception of human thought.

As a species we are particularly prone to a particular type of cognitive bias: what you see is all there is [10]. This cognitive bias forms part of our intuition process by which we come to conclusions based on limited information with the certainty that renders us likely to accept the information as true and, consequently, less likely to seek out more information [10]. As the name of the bias aptly captures, we are inclined to believe that the information we see is a true representation of how the situation is. Even in instances in which subjects were made aware of the limitations in the information they were provided for, the same cognitive bias prevailed [10]. We are so susceptible by manipulation through personalisation algorithms because it seems so compelling to us. We are inclined to accept the information provided to us as 'all there is'. The appeal of personalisation algorithm as a form of attention direction through an appeal to our more subconscious impulses or desires is done in ways we cannot fully see or comprehend. We come to trust the judgements we make are determined partly by our environment, through which we come to accept certain types of knowledge as true, and partly through reality-testing those judgements [10]. Biased information skews one's ability to reality-test judgements we make.

However, Kahneman points out that this environment can be set up in ways that individuals learn the wrong types of things through their experience [10]. For instance, an agent who is relatively well educated(attends school, uses the internet as an informational resource and supplement) in the year 2020 comes across a theory online that the world is flat. She has been taught prior that the earth is a globe with evidence provided to support this assessment. Nonetheless, she is intrigued by this theory and researches it further. As a result, she is shown greater amounts of content relating to theories about the world being flat. She logs off her computer and begins to walk down the road. Her environment does look flat to her like the information she has been ex-posed to online suggests. Her education in the counterarguments evoking the laws of physics and other empirical tests is forgotten. Nor will she be reminded of it in her information searches online. Personalisation algorithms provide us with information in this sort of way. Everything she now sees confirms her belief (the confirmation bias). We are susceptible to taking on as true different opinions should the information provided lead us to that kind of intuition or conclusion. Personalisation, as such, is the perfect tool for propaganda if un-checked.

We design our tools and environments. Machine learning algorithms are coded by human hands after all. For the individual to be autonomous, she must retain the ability to choose for herself her courses of information, protected from coercion or manipulation. The potential to design personalisation in such a way that plays to our strengths and help us mitigate the effects of our flaws, so to speak, is a way that this technology could enable autonomy. Personalisation could be appropriate in instances where it is developed in such a way that is consistent with the values of people who are targeted [19]. The algorithm should transparent which will allow individuals to be aware that the information they are seeing is personalised and factor that into their decision-making process or allow them the knowledge to challenge or change aspects of its functioning [19].

The authors recommend that personalisation should be something that individuals have to consent to given the corresponding use of their data which is often obtained without an individual's consent [19]. Moreover, it is advised that personalisation should not misrepresent reality by either restricting the information that an individual is aware of or by restricting what information can be made available [19].

If we consider the main critiques levelled against personalisation (transparency, choice, and control), these are features that could be resolved. If it were apparent what factors determine the output of personalisation algorithms, it would be less likely that people were influenced or manipulated in ways they were unaware of. This transparency would at least make it explicit that a platform like Google, as an example, uses certain kinds of personal information to inform its user profile, does not include information unrelated to the interests flagged as part of one's profile and prioritises showing relevant content from its partners before non-affiliated information.

Consider again our flat-earther. She has autonomy of the second-order as she can choose an action for herself without influence or coercion that would sway her choice. However, it is dubious that she can question the information she used to guide her beliefs and decisions. Whilst there may be no overt case of manipulation (say someone, somewhere, explicitly decided they were going to feed her information to convince her that the earth is flat) the available information she sees is still controlled in such a sense that seems to misrepresent reality. This is particularly pronounced if she is unaware that she is seeing personalised information based on her past browsing history as she is likely to accept that that what she sees is what there is. As such, she may question the truth of what she is seeing.

Of course, this individual could seek out counterfactuals to verify her argument but most people do not. Especially in cases where people are unaware of what they are seeing is personalised, the view is there is no need for further research. That what they see is all there is. Perhaps a solution is that the efficiency of personalisation quickly producing relevant information to a user is retained but is altered slightly to bring together evidence both for and against topics. Is expanding the results personalisation produces to include counterfactuals to a topic at hand an infringement of their autonomy? No, because there is no force at work that is attempting to stop the user from seeing and interacting with the information that they want to see. There is an attempt to guard against misinformation – whether it be a design flaw of the algorithm or intentionally introduced to manipulate an agent. Such an expansion would give both perspectives on an argument. The user still holds the power to decide what she wants to buy into, but at least the information she is exposed to is diversified in a way that is a new opportunity in the age of information.

Designing personalisation so that individuals are free to see what they want as well as possible alternatives to that perspective may better equip individuals to make their own decisions. Ensuring that as part of that design there is a level of transparency that ensures that the access to individuals granted by the insights generated by personalisation is not used against individuals is to redefine the priority of personalisation. There have been numerous forays into troubleshooting the issues around the use of algorithms. The predominant responses have taken a focus on the importance of accountability and transparency in algorithmic processing. These two facets are considered central to combating some of the negative effects of algorithmic processing. Calls for accountability determine a need to at-tribute responsibility for the consequences or decisions of algorithms to a particular entity or person [12]. This usually is coupled with a call for transparency in which the ways algorithms function provides insight and explanations into how and why it works in the way that it does [4].

Perhaps we could even come to introduce a way to signal that one is looking at information from one source and 'suggest' counterfactuals. We should aim to nudge individuals to consider both sides of a perspective to represent a holistic representation of the topic at hand. There is no way to force one to engage with information one does not want to without violating one's autonomy. However, we can design our systems to facilitate our autonomy. A nudge is not a command. It leaves the individual free to choose whatever it is they wish to believe. But an approach like that would give us an overall representation of issues that we would not alone construct or seek out. One may argue that this kind of determination will result in a lack of autonomy. One could argue that individuals should be free to see the information they want to see. There are two replies to this. Firstly, as our discussion of the current model of personalisation reveals, we may be free to see the information that we want to see but we are not free to question how that information is provided to us, to access alternatives to that which is provided to us and to choose otherwise if there were counterfactuals more appealing. Secondly, just because there is some level of determination at work in how information is recommended or filtered does not alone preclude the possibility of autonomy. It is how things are determined establishes whether or not autonomy is possible. These are all important to establish and further research into technical specifics and potential oversights to implement these kinds of 'enhancements' is still required.

9 Conclusion

Autonomy requires that an individual can question the choices available to her, use that capacity for critical engagement and reflection to evaluate what one's best reasons for action in such a way that is also protected from manipulation. There is always some level of influence at play on any given individual but so long as the individual can autonomously act according to her own reasons rather than someone else's, autonomy obtains. If individuals are to be autonomous, then the systems we rely on for information, such as personalisation algorithms, must be designed to ensure the manipulation of information available to individuals does not happen. As personalisation algorithms function now masses of individuals using platforms on the internet are vulnerable to manipulation. However, the benefits of using algorithms in our everyday lives have been established. As a part of this move, personalisation could be designed to aid us in our flaws as decision-makers and better provide information in such a way that facilitates our ability to deliberate on evidence given for particular reasons or action. This, coupled with transparency, would allow individuals greater ability to question the nature of their world, to draw upon greater levels of information to guide their deliberations on future action and guard against somewhat seem to be our inevitable or unavoidable cognitive errors.

References

1. Ang, C. (2020). Ranked: The Most Popular Websites Since 1993. Retrieved 21 Septem-ber 2020, from https://www.visualcapitalist.com/most-popular-websites-since-1993/
2. Banton, C. (2020). Understanding the Network Effect. Retrieved 22 September 2020, from https://www.investopedia.com/terms/n/network-effect.asp
3. Berry, S. (2020). 2020 Search Market Share: 5 Hard Truths About Today's Market. Retrieved 22 September 2020, from https://www.webfx.com/blog/seo/2019-search-market-share/
4. Binns, R. (2018). Algorithmic accountability and public reason. Philosophy & Technology, 31(4), 543-556.
5. Bozdag, E. (2013). Bias in algorithmic filtering and personalization. Ethics and information technology, 15(3), 209-227.
6. Cadwalladr, C., & Graham-Harrison, E. (2018). Revealed: 50 million Facebook profiles harvested for Cambridge Analytica in major data breach. The guardian, 17, 22.
7. Chambers, C. (2004). Are breast implants better than female genital mutilation? autonomy, gender equality and Nussbaum's political liberalism. Critical Review of International Social and Political Philosophy, 7(3), 1-33.
8. Fuchs, C. (2011). An alternative view of privacy on Facebook. Information, 2(1), 140-165.
9. Hannák, A., Sapiezynski, P., Molavi Kakhki, A., Krishnamurthy, B., Lazer, D., Mislove, A., & Wilson, C. (2013, May). Measuring personalization of web search. In Proceedings of the 22nd international conference on World Wide Web (pp. 527-538).
10. Kahneman, D.: Thinking, fast and slow. 1st edn. Macmillian, New York (2011)
11. Leon, M. (2016). Freedom and Determinism: The Importance of Method. Philosophical Investigations, 39(1), 38-57.
12. Lepri, B., Oliver, N., Letouzé, E., Pentland, A., & Vinck, P. (2018). Fair, transparent, and accountable algorithmic decision-making processes. Philosophy & Technology, 31(4), 611-627.
13. Müller, V. C. (2020). Ethics of artificial intelligence and robotics.
14. Salim, S. (2020). More than six hours of our day is spent online – Digital 2019 reports. Retrieved 21 September 2020, from https://www.digitalinformationworld.com/2019/02/internet-users-spend-more-than-a-quarter-of-their-lives-online.html
15. Schneier, B. (2015). Data and Goliath: The hidden battles to collect your data and con-trol your world. WW Norton & Company.
16. Thaler, R. H., & Sunstein, C. R. (2009). Nudge: Improving decisions about health, wealth, and happiness. Penguin.

17. Williams, J. (2018). Stand out of our light: freedom and resistance in the attention economy. Cambridge University Press.
18. World Internet Users Statistics and 2020 World Population Stats. (2020). Retrieved 21 September 2020, from https://internetworldstats.com/stats.htm
19. Vold, K., & Whittlestone, J. (2019). Privacy, Autonomy, and Personalised Targeting: re-thinking how personal data is used.
20. Tversky, A. & Kahneman, D. (1974). Judgment under Uncertainty: Heuristics and Biases Science, New Series, Vol. 185, No. 4157 pp. 1124-1131.

Part II

AI in Information Systems, AI for Development and Social Good

AI in Information Systems, AI for Development and Social Good: Abstracts of Full Papers Published in Springer CCIS Volume 1342

The paper in this section appears in Springer CCIS Volume 1342 available at https://link.springer.com/book/10.1007/978-3-030-66151-9. The abstract is included in this proceedings.

- Rughbeer, Yastil; Pillay, Anban and Jembere, Edgar. *Dataset Selection for Transfer Learning in Information Retrieval.*

Dataset Selection for Transfer Learning in Information Retrieval

Yastil Rughbeer[1,2][0000−0003−1974−2410], Anban W. Pillay[1,2][0000−0001−7160−6972], and Edgar Jembere[1,2][0000−0003−1776−1925]

[1] University of KwaZulu-Natal, Westville 4001, South Africa
yastil350.rughbeer@gmail.com
{pillayw4,jemberee}@ukzn.ac.za
[2] Centre for AI Research (CAIR), South Africa

Abstract. Information Retrieval is the task of satisfying an information need by retrieving relevant information from large collections. Recently, deep neural networks have achieved several performance breakthroughs in the field, owing to the availability of large-scale training sets. When training data is limited, however, neural retrieval systems vastly underperform. To compensate for the lack of training data, researchers have turned to transfer learning by relying on labelled data from other search domains. Despite having access to several publicly available datasets, researchers are currently unguided in selecting the best training set for their particular applications. To address this knowledge gap, we propose a rigorous method to select an optimal training set for a specific search domain. We validate this method on the TREC-COVID challenge, which was organized by the Allen Institute for Artificial Intelligence and the National Institute of Standards and Technology. Our neural model ranked first from 143 competing systems. More importantly, it was able to achieve this result by training on a dataset that was selected using our proposed method. This work highlights the performance gains that may be achieved through careful dataset selection in transfer learning.

Keywords: Information Retrieval · Ranking · Transfer Learning.

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_4

AI in Information Systems, AI for Development and Social Good: Full Papers Accepted for SACAIR 2020 Online Proceedings

The following full paper was accepted for inclusion in this proceedings. The paper can be cited as indicated below adding page numbers and the url to the specific paper.

- Ndoro, Hakunavanhu; Johnston, Kevin and Seymour, Lisa. *Artificial Intelligence Uses, Benefits and Challenges: A Study in the Western Cape of South Africa Financial Services Industry.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.

Artificial Intelligence Uses, Benefits and Challenges: A Study in the Western Cape of South Africa Financial Services Industry

Hakunavanhu Ndoro ✉[1][0000−0003−0187−9099], Kevin Johnston[2], and Lisa Seymour[3]

[1,2,3]University of Cape Town, Cape Town, South Africa
ndrhak001@myuct.ac.za
kevin.johnston@uct.ac.za
lisa.seymour@uct.ac.za

Abstract. Industries such as agriculture, health, education, transportation, manufacturing, warehousing and logistics are using Artificial Intelligence (AI). The Financial Services Industry (FSI), in particular, benefits significantly from AI because of the massive amount of structured data it consumes and produces. AI is, therefore, transforming the provisioning and management of financial services, with their core business processes being supported by AI. However, current literature has limited coverage of AI uses in the FSI. Hence, this exploratory qualitative cross-sectional timeframe study identified the current understanding, uses, benefits and challenges of using AI in the South African Western Cape FSI. Eleven employees from different organisations were interviewed. It was found that the understanding of AI differs based on the interviewee's background. AI uses in the FSI are found to touch every supply chain activity in the front-office, middle-office and back-office. These uses include predictive analytics, virtual assistants and conversational user experience, and process and application automation. The identified benefits of AI include improved customer experience, customer satisfaction and trust relationship, increased sales, reduced cost and increased efficiency. The identified challenges include lack of training data, bias and lack of skills. Also, the study reveals and confirms social consequences of using AI and suggests future research areas.

Keywords: Artificial Intelligence · Financial Services Industry · South Africa

1 Introduction

Countries such as China and the United States of America are chasing Artificial Intelligence (AI) dominance and making AI a national priority [1, 2]. Industries across multiple sectors are using AI [3]. A research study in AI and related technologies under the banner of AI, such as Machine Learning (ML) and Deep Learning noted significant investments in AI, as well as real benefits and challenges of using AI [4]. The Financial Services Industry (FSI) is using AI solutions to improve financial services such as customer service, decision support and marketing [5, 6]. Although there are multiple conceptions regarding AI, it is increasingly regarded as human intelligence behaviour in computers embedded in business processes, interactions and products [7, 8]. The FSI is reported to be the best industry to use AI because of the massive amounts of structured data it consumes and produces [9, 10]. FSI is seen as the next frontier of AI innovation because of the way AI is transforming the industry's services [5, 11]. The core of many FSI business processes now constitutes AI instead of human actors [12]. A new FSI powered with AI technologies that can manage assets, analyse risk and provide financial advice has been suggested [13]. It is believed that AI enthusiasm in the FSI is currently at its peak [14, 15]. This enthusiasm has been attributed to growth in data and collapsing costs in computing [16].

Yet, the current academic literature in AI has a limited coverage on uses in the FSI [9]. Previous studies of AI use in the FSI have either concentrated on one specific AI application or emphasised specific ML algorithms applied to one FSI problem [17]. The literature on the application of AI in the FSI has focused on investment banking [13]. Therefore, this paper attempted to fill this gap and contribute to the body of knowledge through an empirical study within the context of the Western Cape (WC) of South Africa (SA). WC is considered as a significant financial services hub in SA and it is the largest contributor to the SA Gross Domestic Product (GDP) contributing R43 billion in 2016 [18]. A large proportion of SA insurance, asset management and private investment companies are in the WC [18]. The main research question is: What are the current conceptions, uses, benefits and challenges of AI in the WC FSI? The main research question is broken down into four sub-research questions in the literature review.

The paper provides a foundational starting point for start-up technology providers that are investigating offering AI technology implementations and support to the FSI. Start-ups are significant players in assisting the FSI to determine and identify what AI can do [15]. For SA as an emerging market, AI presents a tremendous opportunity to make more financial services available to more people, including the marginalised and the unbanked that are overlooked by large financial institutions [11, 19]. This paper proceeds in five sections. The next section provides a summary of the literature review, showing how the research questions were derived from the available literature. Section three describes the research methodology by providing details relating to the philosophy of the study, the research approach, method, strategy, data collection, data analysis and ethical considerations. Section four contains the study findings, analysis and discussion. Section five concludes the paper and provided insights for future research.

2 Review of the Literature

There are difficulties in explicitly defining the term AI because of many different definitions of what AI is from different authors of different backgrounds [9, 20].

These different views hinder harmonised international standardisation of what is perceived as AI [20]. The different views have been linked to the complexity of AI as a technology [9]. Four different conceptions of AI were identified and are discussed below. The first conception of AI is that it refers to a set of cognitive technologies with cognitive capabilities [21]. These cognitive capabilities can be categorised into three types; process automation, cognitive insight, and cognitive engagement [21]. Each type of AI links to a business-specific need. Process automation AI links to the need of automating business processes, cognitive insight AI links to the need of gaining insights from large data sets, and cognitive engagement AI links to the need of constant and continuous customer and employee engagement [21]. The second AI conception is from the services provisioning perspective. In this view,AI is equated to machines with human intelligence that can carry out the tasks of humans in the provisioning of services. In this perspective, machines with human intelligence can carry out four types of tasks, that is, mechanical, analytical, intuitive and empathetic [7]. Mechanical tasks can be carried out repeatedly automatically [7]. Analytical tasks are systematic, consistent and predictable tasks [7]. Intuitive tasks include hard thinking, understanding and situation adaption with a high degree of similarity to a human reasoning capacity [7]. Lastly, empathetic tasks are capabilities of machines to feel or behave as though they have human feelings [7].

The third AI conception is from an economics perspective. In this perspective, AI is understood as a modern concept of building rational agents that can perceive their own environment and take actions with the purpose of advancing specified goals [22]. The actions referred to, include making decisions rationally. As a result, a new AI economic system is proposed made up of multiple rational AI agents with capabilities to engage with each other as well as with human beings and firms [22].

The fourth conception of AI is that it is an ML software system with capabilities to learn and adapt to its environment [23]. ML is viewed as a subcategory of AI [24]. ML processes allow computers to change their current programmed functional requirement and act differently according to new information that is made available to them [23]. ML also provides capabilities to analyse large sets of unstructured data [23].

Although this paper acknowledges the identified AI conceptions, we view AI as machines with human intelligence that can carry out the mechanical, analytical, intuitive and empathetic tasks of humans in the provisioning of services [7]. This conception was appropriate for the study since the FSI deals with the provisioning of financial services [25] which can only be provided by machines or humans [7]. Yet local contextual conceptions of AI are not known. Therefore, based on different global conceptions of AI, the first research sub-question arises (RQ1): What is the current understanding of AI in the WC FSI?

2.1 Uses of AI in the Financial Services Industry

The FSI is made up of firms that interact with businesses and households, with the goal of providing financial products and services, such as offering investments, lending, risk management services, wealth management etc. [25]. Common sectors within the FSI includes: (i) depository credit institutions; (ii) holdings and trusts; (iii) non-depository credit institutions and mortgage bankers and brokers; (iv) the securities sector which can include separate divisions within banks and insurance companies, securities brokers and dealers, and registered investment advisers; and (v) the insurance sector [26]. The uses of AI in the supply chain of financial services and products from the FSI to the customer can be put into three categories: front-office, middle-office and back-office [20, 24, 27]. FSI firms are using AI tools across their entire value chain and to all kinds of services and products they provide to customers.

The FSI is using front-office AI applications to interact directly with customers [20, 27]. Retail banks are most likely to benefit from the increased usage of AI-powered tools in front-office applications [46]. Use cases for AI applications in the front-office include the provision of automated customer services with customised engagement using chatbots [28, 29], natural language processing (NLP) and voice assistants [15, 23]. Chatbots are digital virtual customer assistants (VCA) which aid and advise customers via natural-language dialogue on voice or text queries [13, 28]. These VCAs can act on behalf of a customer to perform financial services transactions [30]. Another area where AI is being used in the front-office includes financial identification authentication where customers are identified and granted access to financial institution information systems through face recognition and voice input recognition [10]. An example is the use of AI-driven facial recognition payment systems where a user either smiles at a vending machine [6] or brushes their face [10] to complete payment.

In investment management, AI is being used to offer automated investment management services to investors [25, 31]. These services are in the form of algorithms that emulate human advisors to support investment decisions by investors [32]. AI algorithms are capable of optimising investors' portfolio compositions and making a balanced decision between investor risk tolerance and returns expectation [32]. AI is also being used to automate the decision-making process in lending and credit score applications [6, 33]. In credit scoring, AI decides whether a potential borrower is granted or denied credit [25].

In the back office, AI in the form of robotic process automation (RPA) is being used to automate business processes while reducing cost in delivering those services [34, 35]. Business processes that can be easily automated with AI, mainly with robotics, are those that are repetitive [7, 21]. For example, the application of RPA using natural language processing in automated extraction of information from legal and contractual documents [21, 34]. Other uses of AI for back-office purposes in insurance firms have been increasingly noted [36, 37]. For example, the use of cognitive information in determining the risk associated with individuals, underwriting or generating a risk score [38] and claim processing [24, 37].

In the FSI middle-office, AI is being used by regulators to monitor the financial system better and identify risks [24, 39]. For example, the use of AI in leveraging customer real-time location data to detect and avoid fraudulent transactions [25]. In summary, AI tools could conceptually be divided into three application areas, that is; predictive analytics, virtual assistants and conversational user experience, and process and application automation [40]. Predictive analytics are being applied to any industry domain and across the value chain to help support automation and human decision making [40]. The global use cases of AI in the FSI discussed above and the lack of local contextual studies lead us to the second research sub-question (RQ2): What are the uses of AI in the WC FSI?

2.2 Benefits of using AI in the Financial Services Industry

AI benefits can be viewed as either financial or non-financial [3]. Financial benefits refer to those that directly reduce the costs incurred in delivering a service to a customer, and non-financial benefits are those that can be translated into financial value, for example, increased customer satisfaction [3]. The FSI is using AI to enhance customer experience and help to excel in marketing. For example, the use of chatbots and VCAs in customer service improves customer experience as customers can interact in their own languages with an FSI firm anytime from anywhere using text or voice conversation agents [19, 20, 41]. Intelligent customer service chatbots can achieve higher customer satisfaction rates than live service staff [6]. It is believed that customers trust AI on specific interactions with an FSI, for example, using chatbots because of their inherent absence of self-interest [13]. In FSI marketing, using AI alongside big data sets improves personalisation of services and leads to more relevant product offerings and segmentation, something that cannot easily be achieved by humans alone [11, 34]. For example, an insurance firm may offer home insurance after detecting that a customer is considering buying a new home [27], or if a customer is buying a television, a bank can inform the customer as to the insurance it can provide [13].

Automated decision-making facilitated with AI's higher analytical capacity improves efficiency and enhances financial service quality [19] because AI handles more data (both structured and unstructured), extracting meaning more quickly and more precisely than humans [11, 27]. Decision-making automation uses in credit scoring makes credit applications faster than before [33, 42]. In investment management, Robo-advisors are making wealth management affordable [13, 32]. Therefore, automated decision-making coupled with big-data can increase sales for an FSI firm, since financial services such as credit lending and financial advice that were reserved for a few are now more accessible to people including the unbanked [11, 24, 25, 27].

RPA in automating FSI operations eliminates customer services issues by decreasing business process completion times and removing human keying errors [44], building a strong customer relationship and trust [9, 28]. The overall benefit of RPA in the FSI is operational and human capital cost reduction in the production and provisioning of services and products which in turn results in a lower price for the customer [13, 20, 27]. AI FSI services, compared to human labour, never get tired, never need a vacation and do not require a salary [20], and operate 24/7 [41]. Thus, AI reduces human capital cost by eliminating the need to hire and employ people. Companies offering automated investments solutions at a fraction of the fees of human, financial advisors have been noted [13].

Organisations have been cautioned to not use AI for the sake of AI, but instead, use AI solutions that achieve business goals [13]. AI technologies are valuable tools for FSI firms when implemented for the right reasons and under the right circumstances [46]. Yet SA, with high unemployment, has different conditions to first world countries. From the above literature on AI benefits in the FSI, the third research sub-question (RQ3) arises: What are the benefits of using AI in the WC FSI?

2.3 Challenges of using AI in the Financial Services Industry

Some of the challenges of using AI in the FSI include bias in AI tool decisions, existing and legacy technologies' ability to support AI implementations, lack of human resources and lack of transparency in AI functionalities. The benefits of an AI solution are highly dependent on available training data and prior training on that data [9, 14, 52]. Data training is teaching an AI application to perform assigned tasks accurately by continuously being fed a data set with the aim of improving the success rate of the assigned task [52]. There is a risk of bias in AI decisions if the training data is of low-quality and insufficient, for example, in predicting creditworthiness of a customer [13, 20]. AI reliance on alternative non-structured data sources such as social networks platforms to make decisions is not reliable, as these data sources are inherently biased [20].

Furthermore, humans design and implement AI technologies, and by nature, humans are not perfect [31]. The bias in AI decisions can also be influenced by human data analysts or programmers choosing training data for AI decisions, or the way they program AI algorithms that make AI decisions [13, 20]. For example, algorithms that use consumer's educational level to determine creditworthiness by monitoring their spelling mistakes on internet searches. Therefore, when woman have lower educational levels than men, AI decisions will result in indirect sex discrimination [13]. Thus, low-quality, insufficient and biased training data may lead to unwanted outcomes, such as generating output to the customer that is not within ethical and legal standards [9, 13]. Also, users do not feel that their expectations of what AI should be capable of are being met; for example, a chatbot may not understand user questions or intentions [8]. Although AI can automate most human tasks, there are cases where AI is not able to make complex decisions which require human judgment [13]. The goal of AI should be to empower humans, compensate for human limitations and allow people to expand the possibilities of AI [20, 45]. However, currently, there are shortages of humans with the necessary skills to work with AI tools [20, 46]. Based on this understanding from the global literature on AI challenges in the FSI, the fourth research sub-question (RQ4) arises: What are the challenges of using AI in the WC FSI?

2.4 Literature Summary

AI enthusiasm in the FSI is currently at a peak with AI being used across all FSI institutions. In core business processes, AI is being used in the front office to interact directly with current and potential customers, providing automated customised engagement using digital virtual customer assistants such as chatbots and voice assistants. In the middle and back office, AI is automating manual and labour-intensive processes such as anti-fraud, credit and risk assessments using RPA, automated decision making and predictive analytics. Organisations in the FSI are using AI tools across their entire value chain and for all kinds of services and products they provide to customers. The desired benefits include increasing value for the customer through increased effectiveness as well as reducing cost through increased efficiency. AI is improving the customer experience and presenting FSI customers with new services and capabilities. As with any disruptive technology, AI does have risks and challenges such as bias in AI decisions, training data availability, existing and legacy technologies' ability to support AI implementations, lack of human resources and lack of transparency in AI functionalities. Yet AI uses in the FSI have not been thoroughly studied [9, 13, 17]. In addition, there is limited understanding of the conceptions, uses, benefits and challenges of AI in SA FSI.

3 Research Method

The aim of this study was to expand the current knowledge of AI in the FSI by reviewing the literature and by interviewing, during 2019, eleven individuals working at the top or middle managerial levels for FSI, financial technology (fintech) and AI services provider firms in the WC of SA.

The study made use of the interpretivist philosophy to explore the subjective meanings of persons in the studied domain [43, 47]. The semi-structured interviews [48] used questions adopted from the literature [49, 50]. Due to space constraints, the interview protocol could not be added to this paper; please contact the first author for these. The FSI firms were identified by searching the internet for FSI companies in the WC with heavy use of emerging technologies such as AI to support their business operations. The same process of searching was also used for identifying fintech AI providers in the WC. The main researcher also used his personal connections in identifying organisations.

Due to limited time and the location of the main researcher, the convenience sampling method was used to select participants. Although convenience sampling lacks credibility compared to other sampling methods such as quota, it was necessary for the researcher to get easy access to participants [48]. Ethics approval was obtained by the University prior to data collection. The research participants accepted participation in the study through written consent. Research participants are referred to with non-identifiable codes PX where X refers to a number from 1-11 used. The participants worked in investment management, credit lending, banking, insurance, foreign exchange market and treasury. Included in the sample are firms currently providing AI services such as RPA,

Table 1. Profile of participants.

	Participant role	Organisation type	Main products or services
P1	Data Science Domain Owner	FSI firm	Investment Management
P2	Chief Executive Officer	FSI firm	Wealth Management
P3	Enterprise Architect	FSI firm	Investment, savings, life assurance, asset management, banking, property and personal insurance
P4	Analytics and Optimisation Manager	FSI firm	Personal loans and insurance
P5	Automation Consultant	AI service provider	Robotic process automation
P6	Chief Scientist	AI service provider	Predictive analyses and machine learning
P7	Director	Fintech	Financial services digital experience management
P8	Chief Operating Officer	FSI firm	Treasury
P9	Chief Executive Officer	AI service provider	Cognitive solutions and intelligent automation
P10	Forex Broker	Fintech	Foreign exchange market
P11	Executive Head: Operations and IT	FSI firm	Insurance

predictive analyses, ML cognitive solutions and intelligent automation. Table 1 list all the participants and the respective type of firms they were working in with the corresponding services or products they provide. The study followed the six-phased thematic analysis approach [51] and used a computer-aided qualitative data analysis software tool called NVivo. Thematic analysis has been described as a method that reports on themes within a data set after a thorough identification, analysis, organisation of qualitative research collected data [51].

4 Research Findings, Analysis and Discussion

This section presents the findings from the conducted interviews related to the four sub-questions. Table 2 summarises the AI uses, benefits and challenges that emerged from the collected data. In Table 2, the files column specifies the number of interviews containing the themes. The references column refers to the number of times a theme was coded in the interviews. Also, literature supporting the themes that emerged from the data are listed in Table 2.

4.1 Understanding of AI in the WC FSI

The literature review identified that it is difficult to explicitly define the term AI because of different conceptions of AI from authors of different backgrounds [9, 20]. The difficulty in defining the term AI is one of the themes that emerged

Table 2. Study findings summary.

Themes	Files	References	Related literature
Uses of AI in the FSI			
Risk management	11	28	[24]
Automation	10	21	[9, 20, 27, 30, 34, 35]
Decision making	8	21	[20, 25]
Customer engagement	7	20	[10, 20, 23, 25, 27-29, 36]
Benefits of using AI in the FSI			
Cost reduction and increased efficiency	10	30	[9, 20, 27, 34, 35]
Increased customer experience	9	26	[11, 19, 20, 34, 41]
Improved decision making	4	11	[11, 13, 34, 41]
Financial inclusion	4	10	[11, 15, 19, 24, 25, 27]
Challenges of using AI in the FSI			
Training data	8	18	[9, 14, 52]
Bias	7	16	[9, 13, 20, 31]
Lack of human resources	7	10	[9, 20, 27, 46]

from the data. As P4 said:*"is one of those things that are very difficult to define. I was at a conference a while ago where they have a couple of experts around the world talking about AI, and even they differ in their definition of AI"*. This can also be confirmed by the variations of views given by participants on how they perceive AI. P2 said: *"AI it is basically just a very-very advanced form of technology which actually does a little bit of thinking for you, it allows you to automate more and more because of this higher form of discernment that exists within the technology"*. P1 said: *"I think AI is really just trying to replicate human functions in a computer, so is getting a computer to replicate what humans are capable of, so those will be things like vision and hearing. So, being able to hear a language, understanding audio communication, being able to process that and interpret that into a cognitive response."* The data suggests that in general, the WC FSI does not fully understand AI as it is still in its early days. As P8 said: *"I think it is still very early days and people are caught up in the hype, particularly in South Africa … it is all in the design phase chasing the dragon."*

4.2 Uses of AI in the WC FSI

The literature review showed that the uses of AI in the supply chain of financial services and products span the back-office [20], middle-office [24] and front-office [9, 27]. This study confirmed that AI uses span the full supply chain of the WC SA FSI. Most of the participants believe AI can be or is being used in all the three stages of the FSI supply chain activities. Participants were asked to identify the stage which they believe has the most significant uses for AI within their companies. P9 noted: *"in the front-office in terms of engaging with the client. It is another portal; it is another opportunity for a client to engage with the business without having to use necessarily the expensive costs of an advisor"*. This perception is consistent with [20] and [27] who showed that the FSI is using front-office AI applications to interact directly with customers. Regarding the middle-office P11 stated: *"so that is where we could use AI in the middle-office to work out should I accept the risk, or should I not accept the risk"*. This view supports literature [24] that AI in FSI middle office is being used to identify risk. Regarding the back-office, P2 stated: *". . . in the back-office, the processing is huge, you know, it reduces the need for human intervention, it reduces the errors that a gang form..."* This argument supports the literature that showed AI is used to automate business processes while reducing cost in delivering those services [9, 20, 27, 34, 35].

P5 summarised that uses of AI in the FSI span the entire value chain: *"I think it is initially everyone thought it was back office and middle office, but that is not the case I think with technology coming out today it will be an end to end from receiving that information to fulling that service"*. For the front-office, FSI AI uses, the literature review identified the provisioning of automated customised customer services engagements with chatbots [10, 25, 28, 29, 36], NLP and voice assistants [15, 23]. The uses of chatbots, NLP and virtual assistants were found. When asked in the interview to provide the uses of AI in the FSI, P9 said: *". . . natural language understanding is high so from email automation to a chatbot to the synergy of live chat and chatbot, and in a chatbot that they do not just give you information but can do back end transactional updates, you know, we can actually perform transactions through a conversational interface"*. The back end transactional updates triggered by chatbots referred by P9 supports literature [30] suggesting that virtual assistants, such as chat-bots can execute financial services transactions on behalf of the customer. Other FSI AI use cases suggested by the literature review includes the provisioning of automated investment management services [20, 25].

Opinion among participants differed as to whether automated investment management is currently used in the WC FSI context. P8 said: *"so, we obviously know about Robo-advisor where it started off with a bang, but it is closed very quickly from what I understand"*. Contrary to P8's views, P1 said: *"Other things in the industry there are things like Robo-advisor, is quite popular, you know there is quite a lot happening in that space. So, they deal with trying to build a tool that enables to assess your investment needs as a potential investor. So, instead of going to a person to get advice, you can actually punch in some information into this tool, and the tool will try to provide an assessment of your risk profile and make some recommendations around products"*. The reason for these different views is not clear from the data, but it can suggest that there is a lack of understanding of what the Robo-advisor is.

Regarding the back-office AI use cases, a recurrent theme in the data amongst the participants was the use of RPA. The literature noted that RPA captures and interprets existing FSI business processes, manipulates data, triggers responses and operations to other integrated systems [35]. One of the participants stated that they are currently using RPA in their firm. P3 said: *"so basically using it in our robotics area, and we are also going to use it in some of our services"*. Another theme to emerge in the data was the use of automated decision-making in areas such as credit score applications. P6 said: *"in terms like getting good results for the business the best around credit scoring so people are using machine learning and AI solutions for credit scoring making use of data alternative, for example, bringing social media data into high use where you score someone's credit"*. The literature review mentioned the use of AI automated decisions in credit scoring in determining whether a potential borrower is granted or denied credit [25]. The literature review also indicated the use of data sources such as social networks platforms to make lending decisions [20].

4.3 Benefits of using AI in the WC FSI

In terms of benefits, the majority of participants' views agreed with the literature and identified customer experience as a notable benefit of AI in the FSI. The literature review demonstrated the uses of AI, such as chatbots in customer service and showed how they improve customer experience [19, 20, 41]. As P3 put it: *"We have also moved towards what they call a POCR model, point of contact resolution so in being able to push people to the front. So that we can basically start giving them answers and engagement is a lot clearer on the self-service app or doing it face to face in the branches instead of going to the back-office to investigate all those answers and then going back to the customer. So, we are trying to improve the whole engagement"*. One possible influence of more FSI firms delegating more of the customer interactions to AI applications could be that customers appear to trust AI more because of the absence of self-interest in AI tools [13]. The majority of the participants agreed that AI applications are benefiting the FSI by debiasing decision making. As P6 said, *"you have a very rigid or like statistics and data-based way of making decisions, and you are removing human error, and human bias. So, for example, instead of walking into a bank and somebody does not like your face and decide not to give you a loan, you are removing that kind of bias"*.

Another recurrent theme in the interviews was around the discussion of how AI is improving FSI marketing and enabling cross-selling of FSI products and services. The literature notes that the uses of AI alongside big data sets improve personalisation of services and leads to more relevant product offerings and segmentation [11, 34, 41]. The next two transcript extracts illustrate the views on these benefits: (i) *"in banking most of their work is around how they target customers, targeting different customers to different products, trying to do much work to match customers to the product and obliviously upsell and cross-sell products to you"* (P6). The cross-selling benefit of AI in banking stated by P6 is similar to the literature [13] that if a customer is buying a television, a bank can inform the customer as to the insurance it can provide; (ii) *"so customer 360 is that we have a full view of the customer and that is why we are going to study using outside information to offer better products that are more related so we actually do almost like some homework on the customer and then we also have a whole lot of information already on the customer and remember the customer could have bought products from different segments so now if we can converge the information we can make decisions based on all that information"* (P3). The literature illustrated example of an insurance firm that may offer home insurance after detecting that a customer is considering buying a new home [27] seems to support the P6 example more than the P3 example.

The other benefit of AI in the literature was financial inclusion, making credit lending and financial advice more accessible even to the marginalised thereby increasing the sales of an FSI firm [11, 15, 19, 24, 25, 27]. The financial inclusion benefits also emerged in the data. P6 stated: *"I have seen in financial inclusion are still around credit, and some are around building systems that help people to manage money better, also making sure people have access to finances"*. P8 commented: *"I think there is a space for it in the lower end of the market because people do not have money to pay for someone like me to come and sit with them"*.

The themes of cost reduction and increased efficiency recurred throughout the data as the main benefits of AI uses in the FSI. These themes support the literature findings that the overall benefit of using AI tools such as RPA is to reduce cost and increase efficiency [13, 20, 27, 34]. The following three transcript extracts show how the participants answered in relation to how AI uses can result in cost reduction and increased efficiency benefits: (i) P2, *"it does make things more efficient, and machines are cheaper than people"*; (ii) P4, *"I can touch the thing about costs as well. I mean we can be able to know which customer wants what and do it much more effectively and efficiently using data and AI instead of not using it and shooting blindly"*; and (iii) P6, *"I think the main ones are around cost so like getting solutions that make decisions where you do not have to employ people, which will be cutting down your workforce and getting your computers to do all the hard work around how you make decisions around your customer"*. A reason for the majority of participants confirming that AI reduces cost is that the high human labour cost in the FSI that can be eliminated by AI tools such as RPA [13, 20, 41]. Commenting on cost reduction relating to human labour, P2 said: *"so the most expensive resource in any business like ours is the people"*, and P10 said: *"so it is definitely a cost reduction motivation, it is definitely going to reduce many costs you can think about it if you are an asset management company you have to pay a team of 20 people they cost you R 100 million per year just to have them there"*. This confirms literature that AI-based investment services are offered at a fraction of the cost of human financial advisors [13].

4.4 Challenges of using AI in the WC FSI

The literature notes concern with the integrity of training data [9, 14, 52]. There was a general recognition among most participants that the quality and accuracy of decisions produced by AI are determined by the quality of the underlying data structures. As P10 simply said: *"so data is incredibly important when you make any decisions there is a term which they refer to as garbage in garbage out, so the result of your data is only as good as the input"*. The main challenge around data integrity identified is that the current data quality has not reached the maturity to be used for commercial purpose. P8 said: *"I think everyone is chasing it, everyone wants to use it, but there are serious gaps in the data you* *do not have your data in good enough quality to commercial"*. P6 suggested that this is affecting traditional large banks that have accumulated lots of data in the many years they have been operating. P6 said, *"we have done much consulting work with big banks, things that stand around their data, so if their data is not very well structured"*.

The literature also noted the bias in the automated decisions produced by AI applications due to training data being inherently biased or from bias originating from the designers and implementers of AI system [9, 13, 20, 31]. Half the participants echoed this view and cautioned correct use of data models on which AI solutions are built so that they do not have biases that can be perpetuated in new models. As P10 said: "You have to also check for various biases... you have to make sure that fundamental relationships in the data, especially if you are using past data has not changed... especially if you are making predictions, then you would just be predicting the past might not be relevant anymore".

The reviewed literature also identified the issues of argumented intelligence and shortages of skills in supporting AI implementations, arguing that AI will not completely replace humans [9, 13, 20, 31]. The majority of participants agreed that although AI is a reality, it will not replace some complex problems that require human intervention such as when making a decision that requires personal relationship or empathy. The following four transcript extracts illustrate the participants' views: (i) P11: *"you still need people to configure this, you still need people to teach these things you know, so you still need good people to have the domain knowledge to understand what is it that I look for when I insure a motor car"*; (ii) P8: *"but I do not think that the application of Robo-advice will ever replace the individual advisor"*; (iii) P7: *"my personal view of it is not going to replace people entirely. When you have a query or a real problem, you want to speak to a human. You may want human interaction, chatbots they are good when things are going 100 percent"*; (iv) P10: *"but things like machines cannot display empathy, so if there has been a loss in the family even if you programmed a machine to show empathy I do not think it is going to be real because the person sitting opposite the machine will know that this robot is not being real this robot has been trained"*. The literature notes that AI should not replace humans, but rather enhance human intelligence but also noted the shortages of AI skills in implementing, supporting and working with AI tools [9, 20, 27, 46]. The majority of participants echoed the same view. The participants indicated that there are shortages of AI skills and suggested that AI implementations in the FSI are taking away people's jobs. As P11 said, *"we are short of skills, hugely short of skills in South Africa; IT skills, mathematical skills. We do not have enough, because many of these AI things are based around mathematical principle*" and P10 said: *"I think the financial services industry is going to be affected the hardest this is not just my opinion it is backed by various financial journals as well certain jobs are going to fall away it is not even debatable"*.

5 Conclusion

This study aimed to expand the current knowledge of AI uses in the FSI. Current literature in AI has limited empirical studies on AI uses in the FSI [25]. To achieve this aim, the study identified conceptions, uses, benefits and challenges of AI in the WC FSI. Four research sub-questions were developed from the literature, and eleven participants, working at the top or middle managerial levels for FSI, fintech and AI services provider firms in the WC were interviewed. Data collected from the interviews were analysed to identify themes that linked to the reviewed literature and to the research questions. The study identified that AI uses in the FSI touch every supply chain activity in the front-office, middle-office and back-office. These uses are predictive analytics, virtual assistants and conversational user experience, and process and application automation. The identified benefits include improved customer experience, satisfaction and trust relationship, increased sales and reduced cost and increased efficiency. The identified challenges include lack of training data, bias and lack of skills. Start-up AI technology providers can use the identified AI uses, benefits and challenges as a foundational starting point in determining and identifying what AI can do for the FSI.

This study has limitations. Firstly, the limited availability of AI experts to interview. The majority of the participants were from the FSI firm's top management and their knowledge of what AI is, what it can and cannot do, how it works with humans,and its associated challenges might have been limited. Also, the sample size was small. This study can be expanded to include participants from the economic hub of SA. Taking into consideration the nature of an interpretivist study; the descriptive findings cannot be generalised [40] to the whole of SA, Africa as a continent and other BRICS countries since personal values and viewpoints profoundly impact the data. Another limitation is that, although the main researcher ensured adequate preparation before conducting the interviews, it was his first time to conduct research interviews. Therefore, he lacked research skills and experience.

The findings do provide the following four insights for future research. First, the study confirmed that there are social consequences of job losses because of AI uses in the FSI. Second, the study found that a lack of AI skills is one of the main challenges for FSI firms. These skills challenges need to be addressed. Research is needed on practical guidelines informing governments, higher education and industries, especially in Africa, on the needed collaborations, policies and required reskilling programmes to protect relevant jobs and to develop skills for FSI jobs that are being affected by AI. Third, the AI benefit of financial inclusion requires further research to understand how AI technologies such as NLP, can be deployed in for example chatbots, to cater for all the eleven SA official languages. Fourth, since AI benefits, for example cross selling, rely upon customer data, a further understanding of the effects of AI use on the SA Protection of Personal Information (POPI) Act is required.

References

1. Cave, S., ÓhÉigeartaigh, S.: An AI Race for Strategic Advantage. Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society. 36–40 (2018). https://doi.org/10.1145/3278721.3278780
2. Knight, W.: Trump has a plan to keep America first in artificial intelligence. MIT Technology Review.https://www.technologyreview.com/2019/02/10/137495/trump-will-sign-an-executive-order-to-put-america-first-in-artificial-intelligence (2019). Accessed 22 September 2020
3. Ivanov, S., Webster, C.: Adoption of robots, artificial intelligence and service automation by travel, tourism and hospitality companies – a cost-benefit analysis. In: Proceedings of the 12 International Scientific Conference "Contemporary tourism – traditions and innovations", Sofia University, Bulgaria, 19–21 October 2017
4. Chui, M.: Artificial intelligence the next digital frontier? McKinsey and Company Global Institute. www.mckinsey.com (2017). Accessed 22 September 2020
5. Mehrotra, A.: Artificial Intelligence in Financial Services–Need to Blend Automation with Human Touch. In: 2019 International Conference on Automation, Computational and Technology Management (ICACTM), Amity University, London, United Kingdom. 342–347 (2019). https://doi.org/10.1109/ICACTM.2019.8776741
6. Qi, Y., Xiao, J.: Fintech: AI powers financial services to improve people's lives. Communications of the ACM. 61, 65–69 (2018). https://doi.org/10.1145/3239550
7. Huang, M., Rust, R.: Artificial Intelligence in Service. Journal of Service Research. 21, 155–172 (2018). https://doi.org/10.1177/1094670517752459
8. Svenningsson, N., Faraon, M.: Artificial Intelligence in Conversational Agents: A Study of Factors Related to Perceived Humanness in Chatbots. In: Proceedings of the 2019 2nd Artificial Intelligence and Cloud Computing Conference (AICCC 2019). Association for Computing Machinery, New York, NY, USA. 151—161 (2019). https://doi.org/10.1145/3375959.3375973
9. Kruse, L., Wunderlich, N., Beck, R.: Artificial Intelligence for the Financial Services Industry: What Challenges Organisations to Succeed. In: Proceedings of the 52nd Hawaii International Conference on System Sciences, Grand Wailea, Maui, Hawaii, USA, 8–11 January 2019
10. Zheng, X., Zhu, M., Li, Q., Chen, C., Tan, Y.: FinBrain: when finance meets AI 2.0. Frontiers of Information Technology & Electronic Engineering. 20, 914–924 (2019).
11. Headrick, D., Gobble, M.M.: AI-Powered Fintech Turns Data into New Business. Research Technology Management 62(1), 5–7 (2019)
12. Lin, T.C.: Compliance, Technology, and Modern Finance. Brooklyn Journal of Corporate, Financial & Commercial Law 11(1), 159—257 (2016)
13. Lui, A., Lamb, G.: Artificial intelligence and augmented intelligence collaboration: regaining trust and confidence in the financial sector. Information & Communications Technology Law. 27, 267–283 (2018). https://doi.org/10.1080/13600834.2018.148865
14. van Liebergen, B.: Machine learning: A revolution in risk management and compliance? Journal of Financial Transformation 45, 60–67 (2017)
15. Zhang, X., Kedmey, D.: A Budding Romance: Finance and AI. IEEE MultiMedia. 25, 79–83 (2018). https://doi.org/10.1109/MMUL.2018.2875858
16. Institute of International Finance: Digitising Intelligence: AI, Robots and the Future of Finance. https://www.iif.com/portals/0/Files/private/ai_report_copy.pdf (2016). Accessed 22 September 2020

17. Cavalcante, R., Brasileiro, R., Souza, V., Nobrega, J., Oliveira, A.: Computational Intelligence and Financial Markets: A Survey and Future Directions. Expert Systems with Applications. 55, 194–211 (2016). https://doi.org/10.1016/j.eswa.2016.02.006
18. Invest Cape Town: Financial Services Sector in Cape Town. https://www.investcapetown.com/opportunities/financial-services. Accessed 7 November 2020
19. Lu, C.K.: How AI Fintech Vendors Are Disrupting Financial Services. Gartner. https://www.gartner.com (2018). Accessed 22 September 2020
20. Caron, M.S.: The Transformative Effect of AI on the Banking Industry. Banking & Finance Law Review 34(2),169–214 (2019)
21. Davenport, T.H., Ronanki, R.: Artificial Intelligence for the Real World. Harvard Business Review 96, 108–116 (2018)
22. Parkes, D., Wellman, M.: Economic reasoning and artificial intelligence. Science. 349(6245), 267–272 (2015). https://doi.org/10.1126/science.aaa8403
23. Morgan, R.: The Top Fintech Trends: Driving the Next Decade. American Bankers Association Banking Journal 109(5), 22–27 (2017)
24. Financial Stability Board: Artificial intelligence and machine learning in financial services. https://www.fsb.org/2017/11/artificial-intelligence-and-machine-learning-in-financial-service (2017). Accessed 22 September 2020
25. Gomber, P., Kauffman, R., Parker, C., Weber, B.: On the Fintech Revolution: Interpreting the Forces of Innovation, Disruption, and Transformation in Financial Services. Journal of Management Information Systems. 35, 220–265 (2018). https://doi.org/10.1080/07421222.2018.1440766
26. Phillips L.D.: Exploring the experiences of women of color in contributing to retention and attaining global diversity in the financial services industry: A qualitative exploratory inquiry. Dissertation. Capella University, Minneapolis, Minnesota, USA (2015)
27. Fernandez, A.: Artificial Intelligence in Financial Services. SSRN Electronic Journal. (2019). https://doi.org/10.2139/ssrn.3366846
28. Manser Payne, E., Peltier, J., Barger, V.: Mobile banking and AI-enabled mobile banking: The differential effects of technological and non-technological factors on digital natives' perceptions and behaviour. Journal of Research in Interactive Marketing. 12(3), 328–346 (2018). https://doi.org/10.1108/JRIM-07-2018-0087
29. Okuda, T., Shoda, S.: AI-based chatbot service for financial industry. Fujitsu Scientific and Technical Journal 54(2), 4–8 (2018)
30. Austin, T., Furlonger, D.: Where Banks Can Use Smart Machines. Gartner. https://www.gartner.com (2016). Accessed 22 September 2020
31. Baker, T., Dellaert, B.: Regulating Robo Advice Across the Financial Services Indus-try. Iowa Law Review 103, 713–750 (2018)
32. Day, M., Cheng, T., Li, J.: AI robo-advisor with big data analytics for financial services. In: Proceedings of the 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM '18), 1027–1031. IEEE (2018)
33. Chen, N., Ribeiro, B., Chen, A. Financial credit risk assessment: a recent review. Artificial Intelligence Review 45, 1–23 (2016). https://doi.org/10.1007/s10462-015-9434-x
34. Centre of Excellence in Financial Services: The impact of the 4th industrial revolution on the South African financial services market report. https://www.coefs.org.za/impact-4th-industrial-revolution-south-african-financial-services-market (2017). Accessed 20 September 2020

35. Kumar, K.N., Balaramachandran, P.R.: Robotic Process Automation - A study of the impact on customer experience in retail banking industry. Journal of Internet Banking and Commerce 23(3), 1–27 (2018)
36. Riikkinen, M., Saarijärvi, H., Sarlin, P., Lähteenmäki., I.: Using artificial intelligence to create value in insurance. The International Journal of Bank Marketing 36(6), 1145–1168 (2018). https://doi.org/10.1108/IJBM-01-2017-0015
37. Wilamowicz, A.: The Great FinTech Disruption: InsurTech. Banking & Finance Law Review 34(2), 215–238 (2019)
38. Stempora, J.: System and method for determining an underwriting risk, risk score, or price of insurance using cognitive information. U.S. Patent Application No. 14,463,326, 9 August 2014
39. Chang, S., Shih, C.: The Influence and Application of Artificial Intelligence & Blockchain on Financial Service. HOLISTICA – Journal of Business and Public Administration. 9(3), 45–54 (2018). https://doi.org/10.2478/hjbpa-2018-0022
40. Elliot, B., Andrews, W.: A Framework for Applying AI in the Enterprise. Gartner. https://www.gartner.com (2019). Accessed 22 September 2020
41. Ekholm, J.: How to use AI to create the Customer Experience of the future. Gartner. https://www.gartner.com (2018). Accessed 22 September 2020
42. Vanneschi, L., Micha Horn, D.M., Castelli, M., Popovič, A.: An artificial intelligence system for predicting customer default in e-commerce. Expert Systems with Applications 104, 1–21 (2018). https://doi.org/10.1016/j.eswa.2018.03.025
43. Goldkuhl, G.: Pragmatism vs interpretivism in qualitative information systems research. European Journal of Information Systems 21(2), 135–146 (2012). https://doi.org/10.1057/ejis.2011.54
44. Sturgill, N., Tornbohm, C.: How Bank CIOs Can Use Robotic Process Automation to Improve Customer Experience. Gartner. https://www.gartner.com (2018). Accessed 22 September 2020
45. Sicular, S., Aron, D.: Leverage Augmented Intelligence to Win With AI. Gartner. https://www.gartner.com (2019). Accessed 22 September 2020
46. Sau, M., Brethenoux, E., Cohen, S., Alaybeyi, S., Singh, N.: Cool Vendors in AI for Banking and Investment Services. Gartner. https://www.gartner.com (2019). Accessed 22 September 2020
47. van Zyl, I.: Disciplinary Kingdoms: Navigating the Politics of Research Philosophy in the Information Systems. Electronic Journal on Information Systems in Developing Countries 70, 1–17 (2015)
48. Saunders, M., Philip, L., Adrian, T.: Research methods for business students. Pearson Education, New York (2015)
49. Bäckström, A., Larsson, H.: Is There Such A Thing As Too Much Intelligence? A qualitative study exploring how Born Global e-commerce companies are working towards adopting Artificial Intelligence into their Customer Relationship Management Systems. Bachelor's Thesis. Linnaeus University, Växjö, Sweden (2018)
50. Liliequist, E.: Artificial Intelligence - Are there any social obstacles? An empirical study of social obstacles. Master's thesis, Kungliga Teknisa Hgskolan, Stockholm, Sweden (2018)
51. Nowell, L., Norris, J., White, D., Moules, N.: Thematic Analysis: Thematic Analysis: Striving to Meet the Trustworthiness Criteria. International Journal of Qualitative Methods. 16, 160940691773384 (2017). https://doi.org/10.1177/1609406917733847
52. Smith, D.: What is AI Training Data? https://lionbridge.ai/articles/what-is-ai-training-data (2019). Accessed 22 September 2020

Part III

Applications of AI

Applications of AI: Abstracts of Full Papers Published in Springer CCIS Volume 1342

The papers in this section appear in Springer CCIS Volume 1342
available at https://link.springer.com/book/10.1007/978-3-030-66151-9.
The abstracts are included in this proceedings.

- Baas, Matthew and Kamper, Herman. *StarGAN-ZSVC: Towards zero-shot voice conversion in low-resource contexts.* In Proceedings of SACAIR 2020.
- Jeewa, Asad; Pillay, Anban and Jembere, Edgar. *Learning to Generalise in Sparse Reward Navigation Environments.*
- Kirkland, Lisa-Ann; de Waal, Alta and de Villiers, Pieter. *An Evaluation Framework for Wildlife Security Games.*
- Kouassi, Kouame and Moodley, Deshendran. *An analysis of deep neural networks for predicting trends in time series data.*
- Louw, Johannes. *Text-to-speech duration models for resource-scarce languages in neural architectures.*
- Masha, Ditebogo and Burke, Michael. *Importance Sampling Forests for Location Invariant Proprioceptive Terrain Classification.*
- Pillay, Bradley; Pillay, Anban and Jembere, Edgar. *Hybridized Deep Learning Architectures for Human Activity Recognition.*
- Sooklal, Shivaar; van Zyl, Terence L and Paskaramoorthy, Andrew. *DRICORN-K: A Dynamic RIsk CORrelation-driven Nonparametric Algorithm for Online Portfolio Selection.*

StarGAN-ZSVC: Towards Zero-Shot Voice Conversion in Low-Resource Contexts*

Matthew Baas[0000−0003−3001−6292] and Herman Kamper[0000−0003−2980−3475]

E&E Engineering, Stellenbosch University, Stellenbosch, South Africa
{20786379,kamperh}@sun.ac.za

Abstract. Voice conversion is the task of converting a spoken utterance from a source speaker so that it appears to be said by a different target speaker while retaining the linguistic content of the utterance. Recent advances have led to major improvements in the quality of voice conversion systems. However, to be useful in a wider range of contexts, voice conversion systems would need to be (i) trainable without access to parallel data, (ii) work in a zero-shot setting where both the source and target speakers are unseen during training, and (iii) run in real time or faster. Recent techniques fulfil one or two of these requirements, but not all three. This paper extends recent voice conversion models based on generative adversarial networks (GANs), to satisfy all three of these conditions. We specifically extend the recent StarGAN-VC model by conditioning it on a speaker embedding (from a potentially unseen speaker). This allows the model to be used in a zero-shot setting, and we therefore call it StarGAN-ZSVC. We compare StarGAN-ZSVC against other voice conversion techniques in a low-resource setting using a small 9-minute training set. Compared to AutoVC—another recent neural zero-shot approach—we observe that StarGAN-ZSVC gives small improvements in the zero-shot setting, showing that real-time zero-shot voice conversion is possible even for a model trained on very little data. Further work is required to see whether scaling up StarGAN-ZSVC will also improve zero-shot voice conversion quality in high-resource contexts.

Keywords: speech processing · voice conversion · generative adversarial networks · zero-shot.

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_5

Learning to Generalise in Sparse Reward Navigation Environments

Asad Jeewa[1,2][0000−0003−4329−8137], Anban Pillay[1,2][0000−0001−7160−6972], and Edgar Jembere[1,2][0000−0003−1776−1925]

[1] School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, Westville 4000, South Africa
[2] Centre for Aritificial Intelligence Research, South Africa
asad.jeewa@gmail.com
{pillayw4,jemberee}@ukzn.ac.za

Abstract. It is customary for RL agents to use the same environments for both training and testing. This causes the agents to learn specialist policies that fail to generalise even when small changes are made to the training environment. The generalisation problem is further compounded in sparse reward environments. This work evaluates the efficacy of curriculum learning for improving generalisation in sparse reward navigation environments: we present a manually designed training curriculum and use it to train agents to navigate past obstacles to distant targets, across several hand-crafted maze environments. The curriculum is evaluated against curiosity-driven exploration and a hybrid of the two algorithms, in terms of both training and testing performance. Using the curriculum resulted in better generalisation: agents were able to find targets in more testing environments, including some with completely new environment characteristics. It also resulted in decreased training times and eliminated the need for any reward shaping. Combining the two approaches did not provide any meaningful benefits and resulted in inferior policy generalisation.

Keywords: Generalisation · Curriculum Learning · Sparse Rewards · Navigation.

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_6

Evaluation of a Pure-Strategy Stackelberg Game for Wildlife Security in a Geospatial Framework

Lisa-Ann Kirkland[1][0000−0002−7802−1413], Alta de Waal[1,2][0000−0001−8121−6249], and Johan Pieter de Villiers[1][0000−0003−2506−6594]

[1] University of Pretoria, Pretoria, South Africa
lisakirkland25@gmail.com, alta.dewaal@up.ac.za,
pieter.devilliers@up.ac.za
[2] Centre for Artificial Intelligence (CAIR), Pretoria, South Africa

Abstract Current research on wildlife security games has minimal focus on performance evaluation. The performance of the rangers is evaluated by assessing their game utility, sometimes in comparison with their maximin utility, and other times in comparison with their real-world utility when the game is implemented in a wildlife park. Currently no evaluation framework exists, and this paper proposes an evaluation suite to address this. The movements of the wildlife, the rangers, and the poachers are simulated over a grid of cells corresponding to the wildlife park, where cells containing geographical obstacles are excluded. Poaching and arrest frequency are the primary evaluation measures used. Firstly, we develop a null game to act as a baseline. Typically, one would expect random behaviour of all agents in the null game. However, we simulate random movement for the rangers but more intelligent movement for the poachers. The motivation for this design is to assess whether executing the Stackelberg game yields significantly better ranger performance than random movement, while keeping the poachers' behaviour consistent. The intelligent poachers move by taking their geographical preferences into account and learn from poaching and arrest events. Secondly, we propose that the rangers act as the Stackelberg follower instead of the leader. We formulate a simple pure-strategy Stackelberg game and implement four variations of the game within the framework. The results of the simulations show that the rangers perform better than random when using the Stackelberg game and perform best when acting as the follower.

Keywords: Evaluation · Wildlife security · Game theory · Stackelberg.

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_7

An Analysis of Deep Neural Networks for Predicting Trends in Time Series Data*

Kouame Hermann Kouassi[1,2][0000−0001−9667−260X] and Deshendran Moodley[1,2][0000−0002−4340−9178]

[1] University of Cape Town, 18 University Avenue Rondebosch, Cape Town 7700, South Africa
[2] Centre for Artificial Intelligence Research, 18 University Avenue Rondebosch, Cape Town 7700, South Africa
ksskou001@myuct.ac.za and deshen@cs.uct.ac.za

Abstract. Recently, a hybrid Deep Neural Network (DNN) algorithm, TreNet was proposed for predicting trends in time series data. While TreNet was shown to have superior performance for trend prediction to other DNN and traditional ML approaches, the validation method used did not take into account the sequential nature of time series datasets and did not deal with model update. In this research we replicated the TreNet experiments on the same datasets using a walk-forward validation method and tested our best model over multiple independent runs to evaluate model stability. We compared the performance of the hybrid TreNet algorithm, on four datasets to vanilla DNN algorithms that take in point data, and also to traditional ML algorithms. We found that in general TreNet still performs better than the vanilla DNN models, but not on all datasets as reported in the original TreNet study. This study highlights the importance of using an appropriate validation method and evaluating model stability for evaluating and developing machine learning models for trend prediction in time series data.

Keywords: Time series trend prediction · Deep neural networks · Ensemble methods · Walk-forward validation

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_8

Text-to-speech duration models for resource-scarce languages in neural architectures

Johannes A Louw[0000−0002−8168−7857]

Voice Computing Research Group,
Next Generation Enterprises and Institutions, CSIR,
Pretoria, South Africa
jalouw@csir.co.za

Abstract. Sequence-to-sequence end-to-end models for text-to-speech have shown significant gains in naturalness of the produced synthetic speech. These models have an encoder-decoder architecture, without an explicit duration model, but rather a learned attention-based alignment mechanism, simplifying the training procedure as well as the reducing the language expertise requirements for building synthetic voices. However there are some drawbacks, attention-based alignment systems such as used in the Tacotron, Tacotron 2, Char2Wav and DC-TTS end-to-end architectures typically suffer from low training efficiency as well as model instability, with several approaches attempted to address these problems. Recent neural acoustic models have moved away from using an attention-based mechanisms to align the linguistic and acoustic encoding and decoding, and have rather reverted to using an explicit duration model for the alignment. In this work we develop an efficient neural network based duration model and compare it to the traditional Gaussian mixture model based architectures as used in hidden Markov model (HMM)-based speech synthesis. We show through objective results that our proposed model is better suited to resource-scarce language settings than the traditional HMM-based models.

Keywords: HMM · DNN · Speech synthesis · duration modelling · resource-scarce languages

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_9

Importance Sampling Forests for Location Invariant Proprioceptive Terrain Classification

Ditebogo Masha[1] and Michael Burke[2]

[1] Department of Electrical and Electronics Engineering Science, University of Johannesburg (UJ), Auckland Park, South Africa. 201111866@student.uj.ac.za

[2] Institute of Perception, Action and Behaviour, School of Informatics at the University of Edinburgh, Edinburgh, United Kingdom. Michael.Burke@ed.ac.uk

Abstract. The ability for ground vehicles to classify the terrain they are traversing or have previously traversed is extremely important for manoeuvrability. This is also beneficial for remote sensing as this information can be used to enhance existing soil maps and geographic information system prediction accuracy. However, existing proprioceptive terrain classification methods require additional hardware and sometimes dedicated sensors to classify terrain, making the classification process complex and costly to implement. This work investigates offline classification of terrain using simple wheel slip estimations, enabling the implementation of inexpensive terrain classification. Experimental results show that slip-based classifiers struggle to classify the terrain surfaces using wheel slip estimates alone. This paper proposes a new classification method based on importance sampling, which uses position estimates to address these limitations, while still allowing for location independent terrain analysis. The proposed method is based on the use of an ensemble of decision tree classifiers trained using position information and terrain class predictions sampled from weak, slip-based terrain classifiers.

Keywords: Proprioceptive terrain classification · random forests · importance sampling · autonomous ground vehicles.

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_10

Hybridized Deep Learning Architectures for Human Activity Recognition⋆

Bradley Joel Pillay[1,2][0000−0002−0369−4680], Anban Pillay[1,2][0000−0001−7160−6972], and Edgar Jembere[1,2][0000−0003−1776−1925]

[1] School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, Westville Campus, Private Bag X54001, Durban 4000, South Africa
bradleyjoelpillay@gmail.com
{pillayw4,jemberee}@ukzn.ac.za
[2] Centre for AI Research (CAIR), South Africa

Abstract. Human activity recognition using video data has been an active research area in computer vision for many years. Various approaches were introduced to efficaciously recognize human activities. This study focuses on identifying activities performed by single individuals using visual information from short video clips. Several deep learning techniques are exploited to develop an architecture to effectively solve the human activity recognition task. The architecture hybridizes a two-stream neural network with a multi-layer perception (MLP). The two-stream neural network is a temporal segment network (TSN) which consists of a spatial and a temporal stream. The architecture adopts Octave Convolution neural networks as frame-level feature extractors in the temporal segment network (TSN). The optical flow calculations were performed using the FlowNet 2.0 algorithm, which serves as inputs to the temporal stream. This newly developed architecture was trained and evaluated on the KTH human activity dataset. The results obtained are competitive to existing state-of-the-art results.

Keywords: Human Activity Recognition · Octave Convolution · Temporal Segment Network.

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_11

DRICORN-K: A Dynamic RIsk CORrelation-driven Non-parametric Algorithm for Online Portfolio Selection

Shivaar Sooklal[1][0000−0002−0370−0542] (✉), Terence L van Zyl[2][0000−0003−4281−630X], and Andrew Paskaramoorthy[1][0000−0002−7812−5909]

[1] University of the Witwatersrand,
Johannesburg, South Africa
shivaarsooklal.108@gmail.com, andrew.paskaramoorthy@wits.ac.za
[2] University of Johannesburg,
Johannesburg, South Africa
tvanzyl@uj.ac.za

Abstract. Online Portfolio Selection is regarded as a fundamental problem in Computational Finance. Pattern-Matching methods, and the CORN-K algorithm in particular, have provided promising results. Despite making notable progress, there exists a gap in the current state of the art – systematic risk is not considered. The lack of attention to systematic risk could lead to poor investment returns, especially in volatile markets. In response to this, we extend the CORN-K algorithm to present DRICORN-K – a Dynamic RIsk CORrelation-driven Non-parametric algorithm. DRICORN-K continuously adjusts a portfolio's market sensitivity based on the current market conditions. We measure market sensitivity using the β measure. DRICORN-K aims to take advantage of upward market trends and protect portfolios against downward market trends. To this end, we implement a number of market classification methods. We find that an exponentially weighted moving linear regression method provides the best classification of current market conditions. We further conducted an empirical analysis on five real world stock indices: the JSE Top 40, Bovespa, DAX, DJIA and Nikkei 225 against twelve state of the art algorithms. The results show that DRICORN-K can deliver improved performance over the current state of the art, as measured by cumulative return, Sharpe ratio and maximum drawdown. The experimental results lead us to conclude that the addition of dynamic systematic risk adjustments to CORN-K can result in improved portfolio performance.

Keywords: Online Portfolio Selection · Pattern-Matching · Non-parametric Learning · Systematic Risk

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_12

Applications of AI: Full Papers Accepted for SACAIR 2020 Online Proceedings

The following full papers were accepted for inclusion in this proceedings. These papers can be cited as indicated below adding page numbers and the url to the specific paper.

- Ajayi, Olasupo; Odun-Ayo, Isaac and Leenen, Louise. *On the Performance of Off-Peak Workload Allocation Models in Cloud Computing.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
- Heyns, Austine and Barnard, Etienne. *Word embeddings for recognised multilingual speech.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
- Jeewa, Asad; Jembere, Edgar; Pillay, Nirvana; Singh, Yuvika; Viriri Serestina and Pillay, Anban. *Analysing Coronavirus Trends in South Africa and Countries with a Similar Coronavirus Profile.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
- Mazarura, Jocelyn; de Waal, Alta and Harris, Tristan. emphProbabilistic distributional semantic methods for small unlabelled text. In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
- Mazibuko, Tshidiso and Malan, Katherine. *Machine Learning for Improved Boiler Control in the Power Generation Industry.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
- Mzamo, Lulamile; Helberg, Albert and Bosch, Sonja. *Evaluation of heuristic guided character level word models for morphological segmentation of isiXhosa.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
- Pillay, Nirvana and Jembere, Edgar. *Future Frame Prediction in Transformation Space using goodSTN.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
- Strydom, Rhyno and Barnard, Etienne. *Classifying recognised speech with deep neural networks.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
- Wanyana, Tezira; Moodley, Deshendran and Meyer, Thomas. *An ontology for supporting knowledge discovery and evolution.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.

On the Performance of Off-Peak Workload Allocation Models in Cloud Computing

Olasupo Ajayi[1,2][0000000165833749], Isaac Odun-Ayo[3][], and Louise Leenen[1,2][]

[1] Department of Computer Science, University of the Western Cape, South Africa
3944991@myuwc.ac.za*
[2] Centre for Artificial Intelligence Research, University of the Western Cape
lleenen@uwc.ac.za
[3] Department of Computer Science, Covenant University, Ogun, Nigeria
isaac.odun-ayo@covenantuniversity.edu.ng

Abstract. Cloud Computing is a serviced computing model where computing resources are provided to users on a pay per use basis. Amongst the numerous advantages of this model, cost savings is primary. Effective workload allocation aimed at ensuring adherence to Quality of Service (QoS) and improving resource utilisation is a popular topic in Cloud Computing. It is a NP hard problem, and researchers have employed greedy heuristics such as best/random fit, meta-heuristic techniques such as genetic algorithms, as well as economic models. However, few researchers have directly compared these approaches to determine their relative effectiveness. This paper thus compares five algorithms consisting of economic, heuristic and meta-heuristic models for the allocation of off-peak /batch Cloud workloads. These models are compared on their ability to allocate tasks in a way that effectively utilise Cloud resources, conserve energy, minimise allocation delays and ensure adherence to QoS requirements whilst using real world anonymised data. Our results show that the heuristic approaches excelled in all but QoS adherence, for which the meta-heuristic models performed best. The economic model gave a balanced performance in most of the results. None of the models excelled in all criteria but a weighted formula can be used to select the model that best satisfies a user's preferred metrics.

Keywords: Cloud Computing· Differential Evolution· Genetic Algorithm· Stable Roommate Allocation· Quality of Service· Resource Allocation

1 Introduction

The promises of Cloud Computing are numerous and in truth many have been achieved – especially in terms of cost savings on computing infrastructure. High performance computers with seemingly unlimited storage and processing power are now cheaply and readily available on demand and are accessible from any

* Supported by CAIR.

location, at any time and from any device. However, a potential Achilles' heel to Cloud Computing is user satisfaction. Many factors can lead to user dissatisfaction such as delays, system failure, service disruptions, poor backup systems, long recovery times, poor fault tolerance, capacity limits, etc. Prominent among these is delay, which can emanate from network traffic congestion or a poor task allocation scheme. It therefore comes as no surprise that numerous research works relating to Cloud computing focus on resource allocation and scheduling, with the goal of reducing delays and ultimately improving user satisfaction. However, the focus is often on online user tasks, which require immediate delivery at peak periods; while neglecting batch workloads submitted for off-peak processing during periods of low demand.

1.1 Workload Allocation in Cloud Computing

Workload allocation aimed at ensuring Cloud user satisfaction (QoS) and optimising resource utilisation is an integral part of Cloud computing. To the Cloud users/customers, satisfactory QoS and adherence to Service Level Agreements (SLA) are vital, whilst to the Cloud Service Provider (CSP) effective utilisation of resources and energy conservation are key. These two requirements are often in conflict, as more resources are usually required to guarantee QoS. Balancing these requirements is an intrinsic aspect of Cloud computing and is indeed a NP-hard problem. Researchers have tackled this challenge by proposing effective workload allocation schemes. The rationale is that, if user workloads are effectively matched with appropriate Cloud resources, the requirements of both parties (user and CSP) can be met. In doing this, researchers have considered workload allocation in Cloud computing from different perspectives. One view is synonymous to the bin packing problem [1], where the goal is to put workloads in bins (Cloud servers / physical machines (PMs)) such that a minimum number of bins are used. Another view considers the problem to be a knapsack problem, with the aim of maximising the capacity of each Cloud resource – physical or virtual. Other works have applied statistical models, such as linear regression and historic resource utilisation levels to determine the best way to allocate future workload submissions. Economic and market-based models such as auctioning [2] and stable marriage (roommates) [3] have also been used.

The mentioned techniques apply some form of heuristic algorithm such as best and first fit; 0/1 knapsack and meet-in-the-middle; or regression models. In addition, stochastic approaches have also been applied. Nature inspired meta-heuristic algorithms such as Genetic Algorithm (GA) and Differential Evolution (DE) have mostly been used to find the best mapping of workloads to Cloud resources. Due to their long processing times, these meta-heuristic algorithms are best suited for batch processing of tasks during offline or off-peak periods. Tasks submitted for such off-peak processing are those that are not urgent or for free / freemium users who are satisfied with best effort processing.

1.2 Contributions

To the best of our knowledge, no work has explicitly considered a comparative performance analysis of workload allocation models from diverse domains including deterministic heuristics, stochastic meta-heuristics and economics, when processing Cloud workloads during off-peak periods. Though the authors in [10] proposed allocation models for off-peak tasks, they focused on power utilisation of servers and did not consider allocation models from multiple domains. Similarly, [21] focused on multi-domain allocation schemes for online tasks in federated Clouds.

The contributions of this paper are: i) A comparison of multiple workload allocation models, specifically those best suited for off-peak / batch tasks. The compared models include: Stable Roommate Allocation (SRA) – economic model, GA-based VM Allocation (GAVA) and DE-based VM Allocation (DEVA) – meta-heuristic models. For bench-marking purposes, the popular heuristic models Best-Fit-Descending (BFD) [4] and First-Fit Descending (FFD) [5] are also compared. This comparison is done on the basis of resource utilisation, workload allocation speed, QoS adherence / SLA violation and energy conservation. These metrics are the most considered in the literature, and a summary of metrics used in related works is shown on Table 1. ii) We propose a weighted formula for selecting the allocation model that best suits users' preferences.

This work does not consider workload migration across Cloud servers because once workloads (real time or off-peak/batch) have been scheduled they are considered as tasks within the system and would be processed, migrated, paused/restarted and terminated in a similar manner. The only exceptions are when workload prioritisation are considered, as done in [11, 12], [29], [39]; and this approach has been extensively researched, hence not repeated.

A review of related works is presented in section 2 and the considered models in section 3. A discussion of the various workload allocation techniques that are compared follows in section 4. Section 5 show results and discussions are given in section 6. Section 7 presents a formula for selecting the most suitable scheme. The conclusion and directions for future work are given in section 8.

2 Related Works

Quality of Service. QoS can be defined as the totality of characteristics of a service that bears on its ability to satisfy stated and implied needs of the service user; in essence it is guaranteeing certain performance levels for users, applications and data traffic [6]. QoS are often spelt out in SLAs [7]. In Cloud Computing, QoS adherence is concerned with fulfilling five key requirements - flexibility, performance, reliability, usability and security as they relate to Cloud users [8]. Ran [9] also presented five types of service qualities to be upheld during service provisioning in Cloud environments: infrastructural based QoS attributes -response time, high availability and reliability, robustness and fault tolerance; QoS during deployment - standardisations, change management and stability; security-related QoS adherence - authentication, authorisation, audit/ traceability, confidentiality, integrity, cryptography and encryption; cost/price related QoS attributes; and QoS attributes related to events. IBM [14] identified six Cloud QoS categories: accessibility, adherence to standards, integrity, reliability, response time, and security. In [6], several metrics for measuring QoS are considered: economic, security, performance, availability, scalability, reliability, efficiency, usability and reusability, adaptability, modifiability and sustainability. There is no one rule for defining QoS in Cloud Computing; it seems to a function of perspective and is often abstract or innumerable [7]. This work considers workload allocation time and SLA as metrics for comparing QoS/SLA adherence.

Resource Allocation using Nature-inspired Meta-heuristics. Ferdaus *et al.* used a modified ant colony optimisation meta-heuristic to address resource utilisation and energy conservation [15]. In [18] used a utility function based VM allocation approach to energy conservation, SLA adherence and profit maximisation. They used a GA with expected income minus estimated energy, violation and performance degradation costs as fitness function. In [20] the authors used a Fruitfly optimisation algorithm to address load balancing, delay time and QoS in Cloud Computing. Ref. [19] surveyed numerous Cloud load balancing schemes using various metrics and concluded that GAs performed best across all metrics except for fault tolerance. In [21], a novel GA gene encoding model was used to allocate workloads to Cloud servers; this GA scheme encoded PMs as genes and fitness functions were calculated based on energy consumption. In [12], a model that combined GA and Fog computing was used to improve user experience in Cloud Computing which considered distance, delay and energy consumption. In [26], security was considered as a measure of QoS adherence. They evaluated various security mechanisms and considered the impact of security overheads on QoS and overall system performance.

Resource Allocation using Heuristics. Han *et al.* [22] proposed an approach to workload scheduling based on QoS by introducing a hybrid of Suffrage and Min-Min allocation algorithms and splitting workloads into high and low groups based on their QoS requirements. In [13] the authors approached the problem as a bin packing problem [1] by taking PMs as bins and user workloads as tasks to be packed into the bins. A modified best fit descending heuristics was then used to solve the packing problem. The authors in [16] introduced resource usage prediction based on linear regression. They recorded improved resource utilisation, QoS adherence and energy conservation. Similarly, [11] introduced the binary-search-best-fit algorithm to improve the speed of finding suitable PMs for workloads in Cloud data centres. The authors in [23] proposed a modified algorithm wherein an immune operator and an adaptive scaling factor were introduced into DE to speed it up and prevent premature convergence.

Resource Allocation using Economic Models. Stable marriage, also known as deferred acceptance, has been applied to solve various complex problems. In [28], a framework for matching user requirements (preferences) to Cloud resources using a modified many-to-one stable marriage model is presented.

In [29], the authors presented a practical application of the stable roommate problem to load balancing in global content delivery networks. They modelled network resources as resource trees based on the servers' ability to deliver differentiated services. A modified stable matching algorithm was then used for allocation load balancing. The authors in [2] proposed a market-based resource allocation model for Cloud Computing based on double-sided combinational auctioning. Their model allowed users request a combination of services from multiple competing CSPs.

A number of researchers used hybridised approaches that address the weakness of a given primary method by combining it with another. In [37], the authors combined stochastic Particle Swamp Optimisation (PSO) with a deterministic hill climbing heuristic. They used PSO to find a general solution and hill climbing to improve the local optimal solution. Similarly in [38], GA was combined with heuristic rankings to improve workload completion time.

Table 1. Summary of Related Works

Ref.	Category	Models Considered	Target Objective(s)
[12]	Meta-heuristic (MH)	GA	Conservation of Energy (CoE), Min. allocation delay (delay), Efficient resource utilisation (RU).
[15]	MH	ACO	CoE and RU
[17]	MH	Honeybees algorithm	RU
[18]	MH	GA	CoE, QoS/SLA, Profit
[19]	MH	GA	CoE
[20]	MH	Fruitfly optimisation	delay, RU
[23]	MH	DE	CoE, QoS/SLA
[31]	MH	GA	delay
[34]	MH	PSO	QoS/SLA, delay, RU
[35]	MH	GA	CoE, QoS/SLA
[36]	MH	ACO	Cost optimisation, delay
[37]	MH + Deterministic	PSO + Hill Climbing	delay
[38]	MH + Deterministic	GA + Rank Heuristics	delay
[11]	Deterministic	Binary Search Best-Fit	CoE, CoE, RU
[13]	Deterministic	Bin-packing (BFD)	CoE, delay, RU.
[16]	Deterministic	BFD	CoE, delay, RU.
[26]	Deterministic	Greedy Heuristic	QoS/SLA (security)
[27]	Deterministic	BFD + Round-Robin	delays
[33]	Deterministic	BFD	QoS/SLA
[2]	Economic	Auctioning	RU, profit
[28]	Economic	Stable Matching (SM)	RU, QoS/SLA
[29]	Economic	SM	RU, QoS/SLA
[21]	Multiple	GA + SM	CoE, delay, RU.

Table 1 shows a summary of works in Cloud workload allocation. This paper focuses on three main categories of research: greedy heuristics, meta-heuristics and economic models. For the heuristic approaches, BFD and FFD have mostly been used as benchmarks. Meta-heuristic allocation approaches have mostly employed GA and PSO; stable matching has been the economic model of choice. In this paper, we attempt to compare various workload allocations in Cloud Computing cutting across the different models discussed so far. We consider at least one algorithm from each model type and compare on the basis of efficiency in resource allocation (number of resources used and energy conservation) and user satisfaction (allocation speed and SLA adherence). Descriptive analyses of these metrics as they relate to the models of interest are given in the next section.

3 Model Considerations

Similar to works in [22], [24], [27], this paper considers the allocation of workloads to Cloud resources to be a bin packing problem, and also assume that user workloads are done within Virtual Machines (VMs) which are assigned to PMs. It is also assumed that each VM is fully utilised by its allocated workloads and multiple workloads can be executed on a single VM simultaneously. In the same vain, multiple VMs can be assigned to a single PM. In addition, workload migration between PMs is not considered during the allocation phase, therefore, maximum efficiency is necessary during workload allocation to PMs. For this reason we consider workload allocation to be done in an offline manner. Furthermore, Cloud workload allocations are done with the two objectives of maximising the utilisation of each Cloud resource unit, and minimising the total number of resources used within a Cloud data centre. The first objective seeks to allocate as many workloads as possible to a PM and can be modelled as follows:

Let $W = \{1, 2, ...n\}$ represent a set of user workloads submitted to the CSP for processing and $P = \{1, 2, ...m\}$ represent a finite set of PMs in a Cloud data centre. Eq 1 describes the objective of maximally allocating workloads to a PM, in such a way as to minimise “wastage”; i.e. the total CPU requirements of workloads allocated to a given PM p, is (almost) equal to its capacity.

$$C_p - (\sum_{w=1}^{k} X_w) \to 0 \tag{1}$$

with C_p a PM p's available CPU capacity, X_w a workload w's CPU requirement, k the number of workloads allocated to p, $p \in P$, $w \in W$, and $(0 \leq k \leq n)$.

The second objective ensures that the data centre uses the smallest number of PMs to cater for all received workloads. This leads to better energy conservation and a lower carbon foot print. This process can be modelled by letting a_p be a variable which indicates if a PM, p, is utilised ($a_p = 1$) or not ($a_p = 0$); and b_{wp} be a variable which indicates if a workload w is allocated to $p \in P$ ($b_{wp} = 1$) or not ($b_{wp} = 0$).

The goal is to minimise the number of PMs that are utilised, and is represented by the objective function: $min \sum_{p=1}^{m} a_p$. For offline allocation, both BFD and FFD have been shown to use no more than $11/9 * OPT + 1 bins$ (where OPT is optimal number of bins) [1], [5].

The allocation of workloads to any given PM is subject to ensuring that the maximum capacity of such PM is not exceeded, that is:

$$\sum_{w=1}^{n} x_w * b_{wp} \leq C_p * a_p, \ \forall p \in P, \ a_p > 0 \tag{2}$$

with x_w workload w's capacity requirement, C_p the max capacity of PM p, a_p the utilisation indicator of p and b_{wp} indicating that w is allocated to p.

Once a workload has been allocated to a PM, it cannot be assigned to any other PM. Thus, each workload is completed on only one PM and migration is assumed to be disabled during the initial workload allocation phase. This is given by: $\sum_{p=1}^{m} b_{wp} \leq 1$. Efficient allocation involves utilising the least number of resources (PMs) while maximally utilising each PM without exceeding their capacities. Exceeding a PM's capacity translates to over-utilisation which affects its performance and by extension its ability to satisfactorily service users' workloads (QoS and SLA violations). For SLA violation, we adopt the model proposed in [13] as it has been used by many related works.

With respect to energy consumption, it has been shown that a PM, even when idle, still consumes up to 70% of its maximum usable wattage [25]. Thus for this work, PM energy consumption is taken to be maximum when the PM is being utilised and zero when not in use. The total energy consumption of the data centre is therefore a summation of the energy consumed by all the PMs in executing all allocated workloads. This is modelled as:

$$E_{Total} = \sum_{p=1}^{m} x_w \sum_{w=1}^{n} E_{wp} \tag{3}$$

with E_{Total} the total energy consumed and E_{pw} the energy consumed by each PM $p \in P$ while executing assigned workloads.

Response time is also considered a QoS metric. Factors such as network delays, allocation delays, workload sizes, processing abilities of the PM etc. all influence the response time. However, for the purpose of this paper, only the allocation delay will be considered. To model allocation delay, assume a time limit T_L, beyond which a customer becomes dissatisfied with a Cloud service. Let t be the time it takes to allocate a workload w to a PM p. Assuming that workload allocation are done sequentially and offline, it would take a total of $t * |W|$ to allocate all workloads, where W is the set of workloads to be allocated. The time required to search for the most suitable PMs also has to be considered. Hence, if it takes time s to find the best VM-to-PM match, the total time users have to wait for their workloads to be processed is given by:

$$D = s + \sum_{w=1}^{n} (t * x_w), \ D < T_L \tag{4}$$

with D the allocation delay, x any workload in W and s the PM search time. For each algorithm, s can be described as follows: for SRA, it is the time taken to build the roommates' (workloads) and rooms' (PMs) preference list; for GAVA and DEVA it is the time taken to iterate through the various generations until a global optimal solution (where all workloads are matched to PMs) is obtained; for FFD and BFD it is the time taken to sort the PMs in descending order of processing capacities. Note that the process of finding the best VM-to-PM matching is first completed before actual allocations are done.

Five workload allocation models (BFD, FFD, SRA, GAVA and DEVA) are compared to determine which gives the best overall resource utilisation i.e. the least number of bins (PMs). They are compared on the basis of response time, SLA violation index, resource utilisation and energy conservation. For all metrics, minimal values are desirable.

4 Cloud Workload Allocation Models

Stable Roommate Allocation: SRA is an allocation algorithm that seeks to effectively pair multiple roommates with contrasting preferences for rooms. PMs represent rooms and VMs represent user workloads to be allocated to PMs. Multiple VMs can be assigned to a PM, but each VM can only be assigned to a single PM. SRA is considered for comparison because to the best of our knowledge, beyond auctioning models (which focus on cost and pricing) no other economic models have been applied to workload allocation in Cloud Computing.

Genetic Algorithm-based VM Allocation: The GAVA algorithm proposed in [21] is used with PMs representing genes and a string of genes (chromosome) representing a potential VM-PM allocation solution. Binary encoding was used;if a PM is being utilised it is encoded as 1 and 0 if otherwise. A set of potential chromosomes represent a population. The chromosome with the greatest number of 0s was selected as the best, as this translates to a solution which uses the least number of PMs to serve all VMs.

Differential Evolution-based VM Allocation: Our DEVA algorithm is based on [39]. Similar to the GAVA implementation, in the DEVA each individual also represents a potential match between the VMs and PMs. However, each gene within the individual is equivalent to a VM, with the value of the gene corresponding to the PM to which it was assigned. For instance, if a gene 0 has a value of 4, this means that VM 0 should be allocated to PM 4. At the mutation phase, genes to be mutated are randomly selected, resulting in new potential PM assignments. This further improves the search range and simultaneously prevents premature convergence to a local optimal solution. The crossover process remained similar to that of the standard DE algorithm. These mutation and crossover processes were repeated for several iterations until the best individuals were obtained. DE is often considered as an alternative to GA, it was thus included to benchmark its performance against GA.

Best-Fit Descending and First-Fit Descending: Both BFD and FFD are greedy algorithms used in obtaining an approximately optimal solution for bin packing problem. They have been shown to use $(11/9 * optimalBins + 1)$ bins for offline allocation problems [1]. Though both algorithms work in similar manner, their implementations differ. In FFD, a VM is allocated to the first PM that can accommodate it, while in BFD, a VM is allocated to the PM that best suits it (results in minimal "wasted space"). They are chosen for comparison in this work because they have been used as benchmark in a number of related works (see Table 1) including CloudSim [32].

5 Simulations and Results

Our simulation was carried out using CloudSim with a data centre similar to that used in [11], [16], [18]. A total of 300 simulated PMs were setup in CloudSim, with the PMs being dual core Xeon CPUs clocked at 1,860MHz and 2600MHz, coupled with 4GB of RAM. These were modelled based on the power consumption benchmarks of real world servers as given in [25]. User workloads were executed on single VMs. The VMs were single core, with clock speeds set to either 2,500MHz, 2,000MHz, 1,000MHz or 500MHz. Data used for this experiment were anonymized workload traces submitted to a Google cluster, containing 168 traces measured at regular intervals over seven hours [30].

Allocation Delay: This is a measure of how long users have to wait before processing begins on their submitted workloads as described in section 3. We took allocation delay to be pre-processing time plus time taken to allocate the first task. Figure 1 shows a graphical comparison of the allocation delays for each of the algorithms considered. It can clearly be seen that both FFD and BFD were significantly faster than the other algorithms. Conversely, DEVA was the slowest taking significantly longer to find the best match between VMs and PMs. This is consistent with other reports of the classic DE being general slower to converge than GA [23], [40]. GAVA and SRA were almost at par with an average delay of 21,850,200ns and 19,772,180ns, respectively.

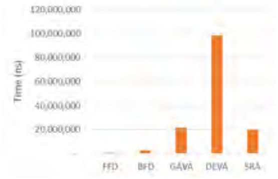
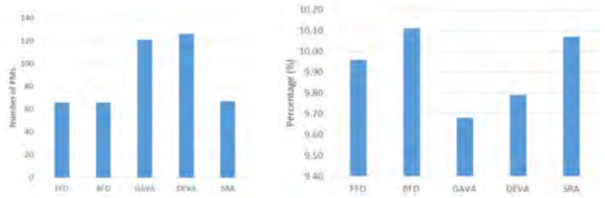


Fig. 1. Allocation Delay

Resource utilisation: This metric is mostly important to the CSP, as lower values translate to less power consumption and lower resource usage (which allows for more room to take on more customers). Results, shown in Figure 2(a), reveal that FFD and BFD packed workloads into the least number of PMs, using only 66 PMs. They were followed by SRA with 67 PMs, while GAVA and DEVA used the most resources at 121 and 124, respectively.

QoS/SLA Adherence: SLAs are used to measure adherence to pre-set user requirements. Since this work focuses on off-peak workloads, it is assumed that



(a) Resource Utilisation

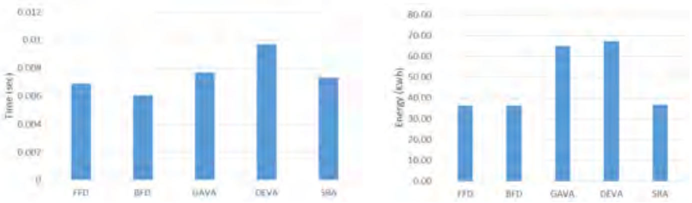
(b) SLA Violation

Fig. 2. Resource Utilisation and SLA Violation Results

users are not particular about quick turn-around time, hence processing times are not considered as a primary measure of QoS. Instead, QoS is taken as a measure of the time during which PMs are operating above a pre-set utilisation index (80%). During these times, it is assumed that the PM is over-utilised and unable to satisfactorily cater for the users requirements.

Figure 2(b) shows a comparison of the average SLA violation of all 5 models. The stochastic models resulted in the least SLA violations with 9.68% and 9.79% for GAVA and DEVA, respectively. FFD follows with 9.96% violation, SRA with 10.07% and BFD at 10.38%. A reasonable explanation for the high SLA violations in BFD is that it seeks to pack workloads into PMs in such a way that gives the least amount of free space. This results in these PMs operating in an over-utilised state for longer periods of time. Conversely, GAVA and DEVA utilise more resources, hence the PMs are less susceptible to being over-utilised.

Execution Time: Though this paper focuses on offline jobs, execution times are still imperative, as they could give insights into the operation time of the data centre. Execution time is taken to mean the total time spent servicing user workloads and lower values are desirable. The results depicted in Figure 3(a) show that job completion times are mostly similar across all algorithms with all being less than a thousandth of a second. BFD allocation resulted in the shortest job completion time at 0.0061s, while DEVA resulted in the longest at 0.0097s.



(a) Execution Times

(b) Energy Conservation

Fig. 3. Execution Time and Energy Comparisons

Energy Conservation: Besides effective resource utilisation, conservation of energy is also very vital to CSPs. There is a global drive to reduce energy consumption and carbon emissions. Comparisons of the five models w. r.t. energy conservation are shown in Figure 3(b). As expected, GAVA and DEVA (at 65KWh and 67.4KWh respectively) consumed the most energy, while FFD and BFD consumed the least at 36.3KWh, followed by SRA at 36.8KWh. These are in line with Figure 2(a), where GAVA and DEVA are shown to utilise about twice the amount of resources used by the others.

6 Discussion

Note that this work focuses on comparing allocation schemes for off-peak Cloud workloads. These are tasks that can be considered non-prior, belonging to users who use Cloud services for free and are mostly satisfied with best-effort services. For most of these allocations models, the workload requirements need to be known a priori, in order to create a matching between workloads and Cloud resources. Table 2 shows a concise performance comparison of the 5 models.

Table 2. Summary of Results

	FFD	**BFD**	**GAVA**	**DEVA**	**SRA**
Allocation Delay (AD)	First	Second	Fourth	Fifth	Third
Resource Utilisation (RU)	First	Second	Fourth	Fifth	Third
Energy Conservation (EC)	Second	First	Fourth	Fifth	Third
QoS/SLA Adherence (QoS)	Third	Fifth	First	Second	Fourth
Execution Time (ET)	Third	First	Fourth	Fifth	Second

When scheduling off-peak tasks, CSPs will be more concerned with utilising the least amount of resources in order to minimise expenses (specifically electricity tariffs), as these tasks yield little or no income. From Table 2 and Figure 2(a), it can be seen that the heuristic models utilised resources best, using only 66 of the 300 servers available in the data centre. These models consequently consumed the least amount of energy. However, their good performance came at the cost of increased QoS /SLA violation. The table also shows that workloads were processed fastest when the heuristic models were used. This is as expected and justifies why these models are used in allocating real time/streaming workloads. The meta-heuristic models on the other hand, resulted in the best QoS/SLA adherence, with GAVA performing the best, followed by DEVA. However, they were slow compared to the heuristic, with DEVA being the slowest of all.

With respect to resource utilisation (and by extension energy conservation), though GAVA and DEVA seem to be the worst performers, the outcome is as a result of the fitness threshold used. For this work, the threshold was set to 40%, implying that only about 120 PMs (of the 300) were utilised. However, it took long hours and thousands of iterations for the algorithms to converge. By using lower thresholds (e.g. 20% or less), GAVA and DEVA could either give results that are at par with BFD/FFD, though at the cost of further allocation delay (and execution time) or never converge.

SRA could be considered a "middle ground"; as it was not as fast as the heuristic models, but much faster than the meta-heuristics. It made up for its lack of speed by being almost at par with the heuristic models in terms of resource utilisation and energy conservation.

7 Selecting the Best Method

Based on the performance results in Table 2 and a user's ordered preferences for the metrics, we propose a weighted formula for selecting the best algorithm:

1. Let $K = \{k1, k2, \ldots k5\}$ represent the set of metrics, such that k1 = AD, k2 = RU, k3 = EC, k4 = QoS and k5 = ET.
2. Let $k_{i-score}$ represent the score of the algorithm under investigation for the metric $\imath$. Table 2 contains a $k_{i-score}$ for $i = \{1, 2, \ldots, 5\}$ of each algorithm. These scores are ranked on a scale of 1 to 5 such that First = 5, Second = 4, . . . , and Fifth = 1. Since no two algorithms have the same score for any given metric, for each metric the sum of its scores is 15.
3. Let $U = \{u_1, u_2, \ldots, u_5\}$ represent the set of user preferences, such that u_i is the user's preference for metric k_i and $\sum_{i=1}^{5} u_i,\ 0 \leq u_i \leq 1$. The weighted formula is given by 5:

$$\sum_{i=1}^{5} u_i * k_{i-score},\ 0 \leq u_i \leq 1,\ 1 \leq k_{i-score} \leq 5 \tag{5}$$

As an example, suppose there are two users with preferences given by $U_1 = \{0.1, 0, 0, 0.45, 0.45\}$ and $U_2 = \{0, 0.4, 0.15, 0.4, 0.05\}$. Using eq. 5, U_1's preferences would result in weights of 3.2, 3.1, 3.4, 2.6, and 3.0 for FFD, BFD, GAVA, DEVA and SRA respectively. For this user, GAVA would be the most suitable to apply. On the other hand, FFD with a score of 3.6 would be best for user U_2.

8 Conclusion

Cloud computing is a serviced computing paradigm whereby resources are provided to users on a pay-per-use basis. Premium users who require immediate and near real-time performance are often prioritised to the detriment of free/freemium users with batch workloads who are satisfied by off-peak processing. In this paper, five off-peak workload allocation models were compared to determine their effectiveness with respect to resource utilisation, energy conservation, QoS adherence and allocation delay. Results show that no particular model excels in all categories; the heuristic models were the quickest and conserved the most energy while the meta-heuristics were very slow but resulted in best QoS adherence. The economic model gave a balanced middle ground performance. A weighted formula that takes user preferences into account, can be used to select the best model for specific users. Though this work has compared multiple allocation algorithm from diverse domains, the comparison was done in a simulated environment. Possible future direction could be in the use of actual test beds such as Open Stack. Furthermore, a hybrid combination of these algorithms, leveraging on their varied strengths could be explored.

References

1. Fernandez de la Vega, W., Lueker, G., Bin packing can be solved within 1 + E in linear time, Combinatorica, 1981, 1(4), 349–355.
2. Fujiwara, I., Aida, K., Ono, I., Applying double-sided combinational auctions to resource allocation in cloud computing. In: IEEE/IPSJ Intl. Symp. on Applications and the Internet, 7-14, (2010).
3. Serrano, R., 2013. Lloyd Shapley's matching and game theory. The Scan. J. of Econ., 115(3),599-618 (2013)
4. Korf, R. A new algorithm for optimal bin packing. In: Aaai/Iaai 731-736 (2012).
5. Yue, M., A simple proof of the inequality FFD (L) 11/9 OPT (L)+ 1,V L for FFD bin-packing algorithm, Acta mathematicae applicatae sinica. 7(4), 321-331 (1991).
6. Odun-Ayo, I., Ajayi, O., Falade, A., Cloud Computing and Quality of Service: Issues and Development, 1, IMECS, p. 6 (2018).
7. Odun-Ayo, I., Ajayi, O., Omoregbe, N., Cloud Service Level Agreements –Issues and Development. In: ICNGCIS, 1-6 (2017).
8. Petcu D., Service Quality Assurance in Multi-clouds. In: Altmann J., Silaghi G., Rana O. (eds) Economics of Grids, Clouds, Systems and Services. LNCS, vol. 9512 (2016).
9. Ran, S., A model for web services discovery with QoS. ACM Sigecom exchanges, 4(1), 1-10 (2003).
10. Hejja, K., Hesselbach, X., Offline and online power aware resource allocation algorithms with migration & delay constraints. Computer Networks, 158, 17-34 (2019).
11. Ajayi, O., Oladeji, F., Uwadia, C., Multi-Class load balancing scheme for QoS and energy conservation in cloud computing. WAJIAR, 17(1), 28-36 (2017).
12. Ma, K., Bagula, A., Nyirenda, C., Ajayi, O., An IoT-based Fog Computing Model. Sensors, 19(12), 2783 (2019)
13. Beloglazov A., Buyya R., Optimal Online Deterministic Algorithms and Adaptive Heuristics for Energy and Performance Efficient Dynamic Consolidation of Virtual Machines in Cloud Data Centers, CCPE, 24(13), 1397-1420 (2012).
14. Mani, A., Nagarajan, A., Improving the performance of your Web services, IBM DeveloperWorks (2002)
15. Ferdaus, M., Murshed, M., Calheiros, R., Buyya R., Virtual Machine Consolidation in Cloud Data Centers Using ACO Metaheuristic, In Euro-Par, 306-317 (2014).
16. Farahnakian F., Pahikkala T., Liljeberg P., Plosila J., Hieu N., Tenhunen H., Energy-Aware VM Consolidation in Cloud Data Centers Using Utilization Prediction Model, IEEE Trans. on Cloud Computing, p. 13 (2016).
17. Randles M., Lamb D., Taleb-Bendiab A., Experiments with Honeybee Foraging Inspired Load Balancing. Proc. of Developments in Systems Engineering (DESE), 240-247 (2009).

18. Mosa A., Paton N., Optimizing Virtual Machine Placement for Energy and SLA in Clouds Using Utilization. JCCASA, 5(1), p.17 (2016). Available at dl.acm.org/citation.cfm?id=3013388
19. Daharwal, P., Sharma, V., Energy Efficient Cloud Computing Vm Placement Based On Genetic Algorithm, IJCTT, 44(1), 15-23 (2017).
20. Lawanyashri M., Subha S., Balusamy B., Energy-Aware Fruitfly Optimisation Algorithm for Load Balancing in Cloud Computing Environments, IJIES, 10(1), 75-85 (2016).
21. Ajayi, O., Bagula, A., Ma, K., Fourth Industrial Revolution for Development:The Relevance of Cloud Federation in Healthcare Support, IEEE Access, 7, 185322-185337 (2019).
22. Han H., Deyu Z., Zheng W., Bin F., A Qos Guided task Scheduling Model in cloud computing environment. Intl. Conf. on Emerging Intelligent Data and Web Tech., 72-75 (2013).
23. Li, J., Liu, J., Wang, J., An Improved Differential Evolution Task Scheduling Algorithm Based on Cloud Computing, DCABES, 30-35 (2018).
24. Nguyen, T. H., Di Francesco, M., Yla-Jaaski, A., Virtual machine consolidation with multiple usage prediction for energy-efficient cloud data centers, IEEE Trans. on Services Computing. (2017)
25. Rice, D., Glick J., Cercy D., Sandifer C., Cramblitt B., Standard Performance Evaluation Corporation (SPEC), 2015, [Online]. Available at www.spec.org/power-ssj2008/results/.
26. Batista, B., Ferreira, C., Segura, D., Leite Filho, D., Peixoto, M. A QoS-driven approach for cloud computing addressing attributes of performance and security, FGCS, 68, 260-274 (2017).
27. Ajayi, O., Oladeji, F., Uwadia, C., Omosowun, A., Scheduling Cloud Workloads Using Carry-On Weighted Round Robin. In Intl. Conf. on e-Infrastructure and e-Services for Dev. Countries, 60-71 (2017).
28. Xu, H., Li, B., Anchor: A versatile and efficient framework for resource management in the cloud, IEEE Trans. on Parallel and Distributed Systems, 24(6), 1066-1076 (2012).
29. Maggs, B., Sitaraman, R., Algorithmic Nuggets in Content Delivery. ACM SIGCOMM Comp. Comm. Review, 45 (3) 52-66, (2015).
30. Wilkes, J., Reiss, C., Google Cluster Usage Traces: format + schema of Google Workloads, 2011, [Online]. Available at http://code.google.com/p/googleclusterdata/
31. Liu, J., Luo, X. G., Zhang, X., Zhang, F., Li, B., Job scheduling model for cloud computing based on multi-objective genetic algorithm, IJCSI, 10(1), 134 (2013).
32. Calheiros, R., Ranjan R., Beloglazov, A., De Rose, C., Buyya, R., Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. Software Pract. Experience 41(1), 23–50 (2011)
33. Banerjee S., Adhikari M., Kar S., Biswas U. Development and Analysis of a New Cloudlet allocation Strategy for QoS Improvement in Cloud. Arabian J. for Sci. and Engr., 40(5), 1409-1425 (2015).
34. Al-maamari A., Omara F., Task Scheduling Using PSO Algorithm in Cloud Computing Environments, IJGDC, 8(5), 245-256 (2015)
35. Sharma, O., Saini, H., Performance Evaluation of VM Placement Using Classical Bin Packing and GA for Cloud Environment, IJBDCN, 13(1), 45-57 (2017).
36. Zuo, L., Shu, L., Dong, S., Zhu, C., Hara, T., A Multi-Objective Optimization Scheduling Method Based on the Ant Colony Algorithm in Cloud Computing, IEEE Access, 3, 2687-2699 (2015). doi: 10.1109/ACCESS.2015.2508940.

37. Negar, D., Navimipour, N., A hybrid particle swarm optimization and hill climbing algorithm for task scheduling in the cloud environments. ICT Express 4, 199-202 (2018). doi:10.1016/j.icte.2017.08.001
38. Keshanchi, B., A. Souri, Navimipour, N., An improved genetic algorithm for task scheduling in the Cloud environments using the priority queues: formal verification, simulation & stat. testing. J. of Sys. & Software, 124, 1-21 (2017).
39. Ma, K., Bagula, A., Ajayi, O., Nyirenda, C. Aiming at QoS: A Modified DE Algorithm for Task Allocation in Cloud Computing." In: IEEE Intl. Conf. on Comm. (ICC), pp. 1-7. IEEE, (2020).
40. Panichella A., A Systematic Comparison of Search Algorithms for Topic Modelling—A Study on Duplicate Bug Report Identification, LNCS, vol. 11664, pp 11-26, (2019).

Optimising word embeddings for recognised multilingual speech

Nuette Heyns[0000−0002−0802−5005] and Etienne Barnard[0000−0003−2202−2369]

Multilingual Speech Technologies (MuST), North-West University, South Africa; and
CAIR, South Africa
{nuette.heyns, etienne.barnard}@gmail.com

Abstract. Word embeddings are widely used in natural language processing (NLP) tasks. Most work on word embeddings focuses on monolingual languages with large available datasets. For embeddings to be useful in a multilingual environment, as in South Africa, the training techniques have to be adjusted to cater for a) multiple languages, b) smaller datasets and c) the occurrence of code-switching. One of the biggest roadblocks is to obtain datasets that include examples of natural code-switching, since code switching is generally avoided in written material. A solution to this problem is to use speech recognised data. Embedding packages like Word2Vec and GloVe have default hyper-parameter settings that are usually optimised for training on large datasets and evaluation on analogy tasks. When using embeddings for problems such as text classification in our multilingual environment, the hyper-parameters have to be optimised for the specific data and task. We investigate the importance of optimising relevant hyper-parameters for training word embeddings with speech recognised data, where code-switching occurs, and evaluate against the real-world problem of classifying radio and television recordings with code switching. We compare these models with a bag of words baseline model as well as a pre-trained GloVe model.

Keywords: embeddings · hyper-parameter · text classification · Word2Vec.

1 Introduction

Distributed word representations or word embeddings are continuous vector representations of words, typically trained on a very large unlabelled corpus. Modern embeddings have their roots in the Continuous Bag-of-Words (CBOW) and Skip-gram vector learning models, known as Word2Vec, as proposed by Mikolov et al. [20]. Since then, word embeddings have been widely used to address natural language processing (NLP) tasks (e.g., language identification, text classification, sentiment analysis etc.) because they encode syntactic meaning as well as semantic relationships between words. Word embeddings are typically applied to written text; in the current contribution, in contrast, we study their application to speech as recognised by an Automatic Speech Recognition (ASR) system. We further restrict our attention to ASR as deployed in South Africa, for reasons that will become clear later in the paper.

South Africa is a multilingual society where people often switch between languages in speech which require custom embedding training techniques. In addition, the embeddings should be derivable from the smaller datasets that are typical of low-resource languages such as the indigenous South African languages. Resource constraints are exacerbated by the fact that code-switched text corpora are hard to find as written materials tend to avoid code switching. While word embeddings are usually trained on large language-specific datasets, we propose to train word embeddings on a South African code-switching dataset, consisting of speech-to-text transcriptions of radio and television broadcast recordings. We calculate the amount of code-switching in the dataset by measuring the I-index (Integration index) introduced by Guzman et al[12]. The code-switched dataset has an I-index of 0.299 suggesting a large amount of code switching. The distribution of the languages in the dataset is calculated using the M-index (Multilingual Index). The M-Index for the code-switched dataset is 0.32 which indicate an uneven representation of the different languages.

The Word2Vec model, still widely used today, is standardised with default values optimised in [20], where the embeddings were used to derive analogies for word pairs. These embeddings can, however, be used on a wide range of NLP tasks. Each NLP task has peculiar characteristics and challenges; hence, it is important to configure the hyper-parameter settings to fit the needs of the specific task of interest. Our main contribution is to show how the hyper-parameters of the Word2Vec model should be chosen so that the model can be applied in the context of recognised multilingual speech. Using Word2Vec, this study evaluates embeddings resulting from different hyper-parameter modifications to identify which hyper-parameters should be adjusted and in what way.

There are two approaches to evaluate word embeddings; intrinsic and extrinsic tasks. Intrinsic tasks (e.g. similarity and analogy tasks) are widely used to test the quality of word embeddings, however, these tasks do not always correlate to real-world applications. Extrinsic tasks test how well an embedding performs on a real-world application. We propose to extrinsically test the performance of the models trained with different parameter settings by applying it to text classification. Because we are working with television and radio broadcasts, it is natural to classify the data into five categories namely news, advertisements, sport, traffic and weather. We found that optimising the hyper-parameters can produce a 31% performance increase on a text classification task. When comparing the code-switched model to a Bag of Words (BOW) baseline model and a pre-trained GloVe model, we found that the optimised Word2Vec model outperformed the pre-trained models and showed results that are competitive to the BOW model. These findings show the importance of optimising the relevant hyper-parameters to fit the task at hand.

2 Related work

Most research done on bilingual word embeddings train word embeddings using monolingual datasets. Pratapa et al. [21] argue that using monolingual datasets is not sufficient when dealing with a multilingual environment because monolingual datasets do not represent the syntactic structures and cross-lingual semantic associations present in a code-switched dataset. Pratapa et al. [21] used a synthetic code-switched dataset generated using linguistic models to train their word embeddings. We propose to go one step further by using recognised speech data that contains natural code-switching, as our dataset.

Optimising the hyper-parameters of word-embeddings is known to be important. Li et al. [17] compared different word embedding architectures and found that when all hyper-parameter settings are standardised across the different architectures, they all perform very similarly. Thus, the specific embedding architecture is not of major importance, as long as the hyper-parameters are optimised for the specific tasks of interest. For our study, we will be investigating the importance of different hyper-parameters and their optimum value for Word2Vec.

According to Faruqui et al. [8] embeddings should be tested on the specific task it will be used for as embeddings can capture different information depending on their parameter settings and the dataset used to train them. A couple of studies Chiu et al. [5], Schnabel et al. [23], Linzen [18] and Gladkova and Drozd [11] investigated the relationship between intrinsic and extrinsic evaluation methods. These studies all concluded that there is no direct correlation between the performance of an embedding on intrinsic tasks and their performance on a real-world problem. In this study, we will be testing our embeddings with an extrinsic evaluation method.

Machine learning algorithms such as decision trees, Naive Bayes and Support Vector Machines (SVMs) are popular text classification methods. The input is usually represented as a BOW where each word is represented by a single dimension and the dimensionality of the vector is the size of the vocabulary. Not only does a BOW model have high dimensionality and a sparse feature matrix, but BOW models do not represent the relationship between words - neither syntax nor semantics. The semantic information in a document can be used to differentiate between text that is using the same vocabulary to express different ideas on the same subject. This is especially useful when classifying text according to sentiment. There are a few techniques used to enhance the input in order to include information about the semantic relationships between words or phrases. Probabilistic latent semantic analysis (PLSA) and Latent Dirichlet Allocation (LDA) are methods that create a low-dimensional space where concepts are represented by multiple expressions. NLP methods, such as Named Entity Recognition (NER), Part of Speech Tagging (POS-tagging) and semantic role labelling as well as sources like WordNet [9] and Wikipedia can also be used to enrich the dataset. These features are usually still sparse and redundant information exists among these features [13]. Bojanowski et al. [2] proposed a simple method to update the word vectors to fit the distribution of a new dataset in order to extend the lexicon by combining low frequency words from different datasets. As stated by the authors, this method is not a definitive solution and rather serves as a proof of concept. Sinoara et al. [24] used pre-trained vectors to obtain knowledge enhanced document embeddings. Huang et al. [13] proposed a neural network architecture to classify text documents. Each document is represented by a low dimensional embedding similar to word embeddings. For this method, the word embeddings were learned directly in the text classification task because they argued that learning the embedding in a separate step is not optimal for text classification. Le and Mikolov [16] used paragraph Vectors where embeddings can be derived from text varying in length ranging from sentences to documents.

3 Methodology

In the following section, we will discuss the different hyper-parameters that usually have an effect on the embedding quality. When reviewing the experimental setup we will discuss the dataset, models and the hyper-parameters that we will be fine-tuning. Following the experimental setup, we will discuss the evaluation metrics and compare the different models.

3.1 Word embedding hyper-parameters

When creating a word embedding model there are a few default parameter values that should be fine-tuned to fit the evaluation task as well as the dataset. These parameters include embedding size, window size, training time, minimum count, learning rate, negative sampling and the negative sampling distribution. It is necessary to identify which of these parameters is important for us to fine-tune by looking at the characteristics of each of the parameters, the dataset and the evaluation task.

Embedding size The embedding size is the number of dimensions in the embedding layer of the network. If the embedding dimension is higher than the vocabulary size, it will lead to over-fitting because there is no cross-word interference and the embedding would resemble a one-hot-encoding more than a true embedding. The embedding size should therefore correlate to the dataset size. A smaller dataset will perform better when using a smaller embedding size and vice versa. Caselles-Dupré [7] tested their embedding size on three different datasets varying in size and observed similar results with all three dataset sizes. The performance on their tasks stopped showing improvement when the embedding size was increased to 200. These findings are in contrast to the theory that the embedding size should correlate to the dataset size as the same embedding size was optimal for all three dataset sizes. The task the embedding is used for can however also play a role e.g. text classification uses fewer dimensions than sentence generation.

Window size The window size is the number of context words around the target word. Research has indicated that using a smaller window size (e.g. 2 to 15), the clusters will show interchangeable terms, so synonyms and antonyms are clustered together. Fanaeepour et al. [7] tested their embeddings on similarity and analogy tasks and found that a window size greater than four has no significant impact on the performance of their tasks. With a larger window size, the clusters are expected to show the relatedness of words which is preferable with tasks like text classification. Some researchers have stated that the window size does not make a difference when using a larger dataset.

Training time The number of times the algorithm iterates over the training data is known as the number of training epochs. The default number of epochs for word embedding models is 5. Increasing the training time can be computationally expensive especially with larger datasets.

Minimum count The model ignores words that occur less than the minimum count. For large datasets this is less important.

Learning rate Dillenberger [6] found that a higher learning rate over a longer time is beneficial for rare words but that a learning rate decay ultimately creates better word embeddings. Research shows that the range of the learning rate usually lies between 0.05 and 0.5

Negative sampling To optimise the Word2Vec model, negative sampling is used. With negative sampling, the calculations of the loss function are only performed on a subset of the input and thus speeds up the process [6].

Negative sampling distribution The exponent is used to shape the negative sampling distribution. A uniform distribution of α=1.0 will sample words in proportion to their frequencies. A unigram distribution of α=0 will sample all words equally, and a negative α value will sample low-frequency words more than high-frequency words [4]. The default value of α=0.75 is set according to experiments in the original Word2Vec paper of Mikolov et al. [20] where the parameters were tested on intrinsic tasks. Caselles-Dupré et al. [4] suggest that other values may perform better for recommendation applications.

3.2 Experimental setup

Data For our South African code-switching dataset, we used speech-recognition outputs produced by a commercial speech-recognition system developed by Saigen[1], on South African radio and television broadcasts obtained from Novus[2]. The dataset includes recordings of 103 different South African radio and television stations. On these stations 12 different languages are spoken; English, Afrikaans, isiZulu, isiXhosa, Sesotho, Tshivenda, Sepedi, Siswati, Setswana, Ndebele, xiTsonga and Hindi. 77 Of these stations are in English, 8 in Afrikaans, 6 in Setswana, 4 in isiZulu and the rest of the languages each have 1 dedicated station. For the purpose of this study only the English radio station data was used. This data however still includes much code switching to the other languages and is, therefore, a useful representation of spoken South African English. Simple pre-processing was done using Gensim's pre-processing library. The data was lowercased, tokenised, lemmatised and stop words were removed. After pre-processing the dataset consisted of 100K words, which were split into a training and test set of 70K and 30K respectively.

[1] `https://www.saigen.co.za/`

[2] `http://novusgroup.co.za/`

Rijhwani et al.[22] used word-level language identification to discover code-switched sentences on Twitter. They used the number of switch points in a tweet to indicate the degree of code switching. For example, English-German tweets generally have only one switch point that implies the tweet is usually only a translation of the same content and not an example of true code switching. Gambäck and Das [10] argued that as the level of code switching increases in a dataset, the performance of the model is expected to decrease. There are several statistical measures to compare the level of code switching in datasets. In this study we used the M-index and the I-index.

1. M-index: Introduced by Barnett et al. [1], the Multilingual Index (M-Index) quantifies the ratio of languages in the corpus based on the Gini-coefficient. It measures the degree in which a language is distributed in the dataset, however the M-Index does not indicate if the languages are integrated with each other and therefor it is not proficient for indicating whether code switching occurred or not. The M-index are calculated by the equation
$$M - Index = \frac{1 - \sum_j p_j^2}{(k-1)\sum_j p_j^2} \tag{1}$$
where $k > 1$ is the number of languages, p_j is the number of words in a language over the total number of words in the dataset. j range over all the languages present in the dataset. The M-index is bounded between 0 (a monolingual dataset) and 1 (where each language in the dataset is represented equally)
2. I-index: Introduced by Guzman et al.[12], the Integration index complements the M-index by summing up the probability that there has been a switch. It serves as a simplified version of the revised CMI index of Gambäck and Das [10]. A valuable benefit of the I-Index is that it does not require dividing the dataset into utterances or for it to contain computing weights. Given a dataset where each token has a language tag $\{l_i\}$ where i ranges from 1 to n, the size of the dataset:
$$I - index = \frac{1}{n-1} \sum_{1 \leq i < i+1 \leq n} S(l_i, l_{i+1}) \tag{2}$$
where $S(l_i, l_{i+1}) = 1$ if $l_i \neq l_{i+1}$ and 0 otherwise. The index is bounded between 0 (where no switching occurs) and 1 (if code switching occurs between each word pair.)

To measure the M-index and the I-index, each word in the corpus has to be tagged with their language. Whatlang is an extension of fastText's language identifier model that can identify languages on a word level. Custom models for the South African languages were trained and used to identify each word in the corpus and tag them with the LID. The number of different languages that was identified is summed. The number of times a language switch occurred in the data was calculated using the I-index. The I-index for the code-switched dataset is 0.299, suggesting a large amount of code switching. The distribution of the languages in the dataset was calculated using the M-index. The M-Index for the code-switched dataset is 0.32 which indicate an uneven representation of the different languages

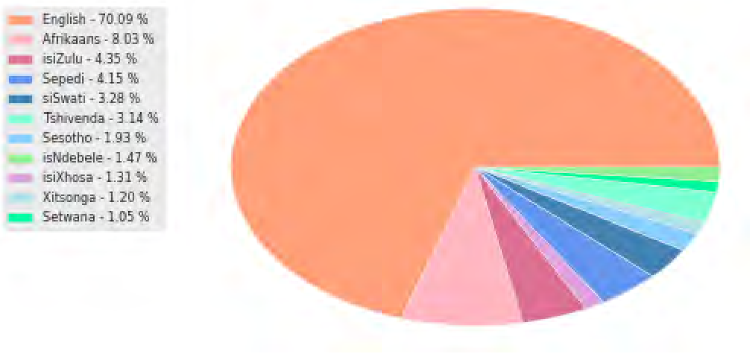


Fig. 1: The number of words in each language included in the code-switched dataset.

Models Word2Vec is based on the distributional hypothesis that words used in the same context, will have similar meanings. There are two methods proposed by Mikolov et al. [20] when training Word2Vec models; CBOW and Skip-gram, both of which consist of a single-layer architecture where the inner product of two-word vectors are calculated. A Skip-gram model aims to predict the context, given a word, while CBOW's aim is to predict a word, given the context. According to Mikolov et al. [19], the CBOW model performs better with news data and less data is needed than with skip-gram. Kim et al. [14] also claims that the accuracy of a CBOW model is higher and more stable compared to that of a Skip-gram model when the model is evaluated by classifying news articles. A CBOW method was used to train the custom Word2Vec models for which different hyper-parameter settings were experimented with. Word embeddings cannot be directly used to train a text classifier as documents have words with various lengths. That is why a weighted average of all the words in the document has to be used. Each of these models was therefore averaged with Mean Square Error (MSE) and Term Frequency–Inverse Document Frequency (TF-IDF) respectively.

For a baseline model, we implement a simple BOW model for which the parameters have been optimised. The baseline model ignores the ordering of relationships within the text. The document is translated to a vector of term weights where each dimension corresponds to a term and each value represents the relevance. We will also be comparing our Word2Vec models with a pre-trained GloVe model that is again averaged with MSE and TF-IDF respectively.

Hyper-parameter selection The hyper-parameters we explore in this study are listed in Table 1.

Table 1: The variable testing range for each hyper-parameter. Highlighted in red are the default values for each hyper-parameter.

Hyper-parameter	testing range
Embedding size	50, 100, 150, 200, 250, 300
Window size	2, 5, 10, 15, 30, 40, 50
Training time	2, 5, 10, 20, 40, 80, 160
Minimum count	0, 2, 4, 6
Learning rate	0,0025, 0.025, 0.25
Negative sampling	0, 5, 10, 15, 20
Negative sampling distribution	-0.5, -0.25, 0, 0.25, 0.5, 0.75, 1

4 Evaluation

An extrinsic evaluation method is used to evaluate the word embedding models. The word embedding models are used to create a text classifier that classifies the text into the categories news, sport, traffic, advertisement and weather. A logistic regression classifier is used to classify the text. The performance of a classification model depends on the quality of the training data and the quality of the representation model. The dataset was tagged with the appropriate categories. The classes are very imbalanced; 38,02% of the data is sports data, 36,49% of the data is news data, 14,74% is advertisements, 6,96% is traffic reports and 3,78% is weather data. The embeddings are expected to perform better with the high resource classes.

4.1 Comparing the feature selection models

Each of the Word2Vec models trained with different hyper-parameters was classified using a logistic classifier. Precision-recall (PR) curves are used to assess performance given the unbalanced datasets [3]. Micro-averaging considers each element of the class indicator matrix as a binary prediction, whereas macro-averaging gives equal weight to the classification of each class. because our dataset is imbalanced, we do not want to give equal weight to each class and will therefore be looking at the micro average PR curve of each model. For the initial test, each hyper-parameter was changed in isolation so that the effect that it has on the default model can be observed. The biggest improvement in the model's performance was seen when negative sampling was used. A small

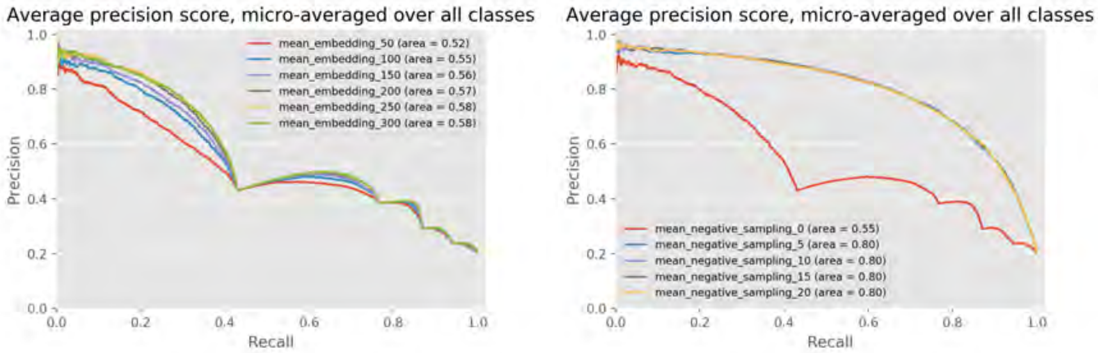
Table 2: Examples of sentences in the dataset tagged with their appropriate catagories.

Tag	Sentence:
News	in limpopo the body of the deceased was found next to the road in the bush buck ridge area last week police spokesperson leonard tladi says the mother appeared in court yesterday also on a charge of murder
Sport	the southern kings go in search of their second win of the season against conduct while the cheetahs take aim at...
Traffic	inbound slows down between sable road and the elevated freeway traffic lights are faltering retreat at prince george drive and military road
Advertisement	with our easy to use guide dstv keeps you up to date with latest episodes and action from us connect to your explorer just schedule recordings so you never miss download the app and keep the world at your fingertips
Weather	it's around twenty-one degrees in joburg good morning

sampling rate of 5 was enough to increase the model's performance by 25%. The PR curve is smoother and has monotonic attributes when negative sampling is used. The model showed a 6% increase in performance when the embedding size was increased to 250. After the embedding size was increased to 250, the model stopped showing further improvement, comparable to the findings of Fanaeepour et al. [7], who did not observe any improvements when the embedding size was increased beyond 200. The experiments showed that changing the window size, training time and minimum count in isolation has no effect on the model. Figure 2 shows the micro-averaged PR curve for the models trained with different embedding sizes and negative sampling rates. The models averaged with MSE performed slightly better than the models averaged with TF-IDF. To assist the readability of the graphs, only the models averaged with MSE are included.

4.2 Comparing the interactions of multiple parameters

Krebs and Paperno [15] showed the importance of the interaction of different hyper-parameters. Their research focused on three hyper-parameters; subsampling, shifted Pointwise Mutual Information (PMI) and context distribution smoothing. Their work showed that the same performance can be achieved when using datasets of different sizes, as long as the combination of hyper-parameters is optimal. We continued our hyper-parameter optimisation by looking at the effect that some of the hyper-parameters have on each other. The window size, training time, minimum count and learning rate had no effect on the model's performance when varied in isolation. Since there are too many combinations to exhaustively test all interactions, we discuss five combinations that were found



(a) Changes in the embedding size

(b) Changes in negative sampling

Fig. 2: Micro-averaged precision-recall curves for different embedding size and negative sampling for feature selection models

to interact significantly. We found that changing the negative sampling in combination with other hyper parameters can have a significant effect on the model. The only hyper-parameters that do not show an effect when changed in combination with negative sampling are window size and minimum count. Changing the window size in combination with minimum count gives a slight improvement of 1% that resulted in the best performing model for our task.

Learning rate + negative sampling when the learning rate is changed in combination with negative sampling, a difference of up-to 10% is reached.

Learning rate + negative sampling distribution Using the default smoothed unigram distribution where $\alpha = 0.75$, the model will benefit by a smaller learning rate of 0.0025, but ultimately the model will perform better when using a unigram distribution where $\alpha = 0$ in combination with a higher learning rate of 0.025.

Negative sampling + negative sampling distribution We found that when changing the negative sampling rate to 5 and using a unigram distribution where $\alpha = 0$, the performance increased with an additional 2%. This combination continued showing the best performance independent of how other hyper-parameters were changed.

Training time + negative sampling The optimum time to train the model when negative sampling is optimised, is 10 epochs, after which the model's performance starts to decrease.

Window size + minimum count When the window size is 30 or higher, all minimum count values achieve the same performance. Overall, changing the minimum count and window size do not show significant performance differences in the model.

4.3 Evaluating the code-switched model model

The values that performed the best for each hyper-parameter combination were used to train the code-switched model. Table 3 compares the default Word2Vec hyper-parameters with the optimised values.

Table 3: The default and optimised hyper-parameter values.

Hyper-parameter	Default value	Optimised value
Embedding size	100	250
Window size	5	15
Training time	5	10
Minimum count	2	2
Learning rate	0.025	0.025
Negative sampling	0	5
Negative sampling exponent	0.75	0

For our task we found that increasing the embedding size, window size, training time and negative sampling rate, creates a better model. The default learning rate performed the best of all learning rates and changing the minimum count does not have any significant impact on the model, and the default value can be kept. Figure 3 compares the performance of the default hyper-parameter values with the optimised hyper-parameters of the Word2Vec model. The model showed a 31% improvement on the default values when the values are optimised.

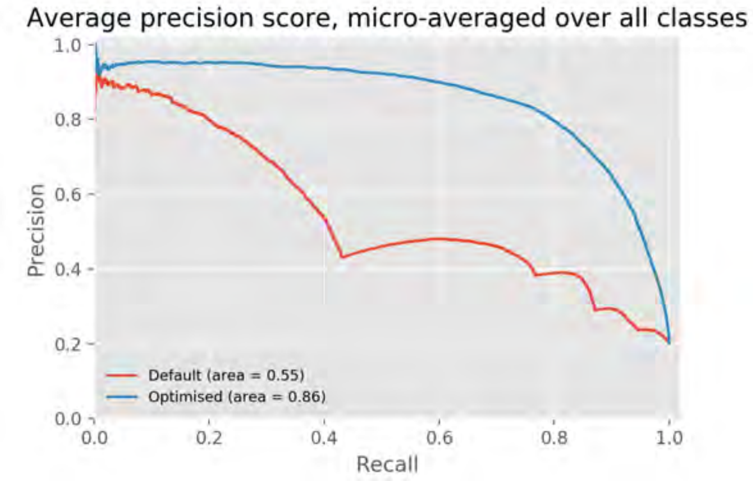


Fig. 3: Micro-averaged precision-recall curve for the default values vs the optimised values

4.4 Comparison between baseline and feature selection models

We compare the default as well as the optimised Word2Vec model (code-switch model) with a BOW baseline model as well as a pre-trained GloVe model. Figure 4 shows the precision-recall curve for the 4 different models. The code-switch model shows competitive results, whereas the default model is outperformed by both the pre-trained GloVe model, as well as the BOW model. The performance of the pre-trained GloVe model demonstrates the importance of a) optimising the model's hyper-parameters to suite the task, and b)the dataset should be appropriate for the application environment. The pre-trained GloVe model was trained using the Google News dataset, a dataset that only includes English examples. The fact that our relatively small code-switched dataset performs better than a similar embedding pre-trained on a much larger monolingual dataset suggests that training embeddings on a code-switched dataset when it will be used in a code-switched environment is more important than the size of the dataset the embedding will be trained on.

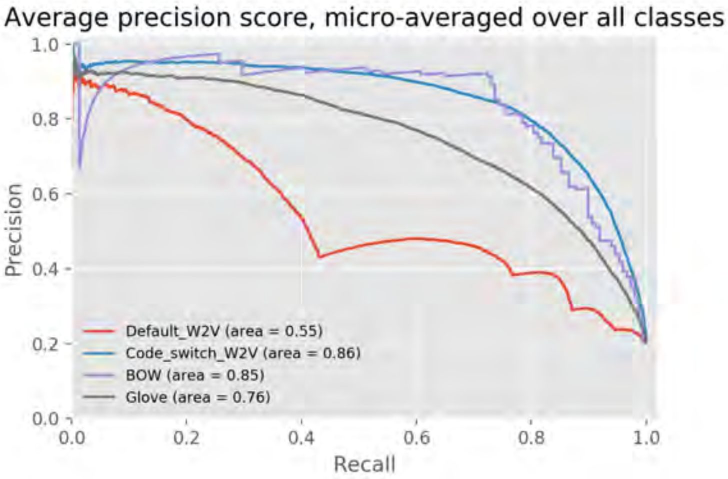


Fig. 4: Comparison between baseline and feature selection models.

5 Conclusion

Word embeddings are popularly used to address a wide range of NLP tasks because they encode syntactic meaning as well as semantic relationships between words. The Word2Vec architecture is a popular choice to train these models. Word2Vec is standardised with default hyper-parameter values optimised in the original research [20], where the embeddings were used to derive analogies for word pairs. Each NLP task, however, has it's own characteristics and challenges. This study evaluated the effect that fine-tuning certain hyper-parameters has on the model's performance when evaluated on an extrinsic text classification task. We found that fine-tuning the embedding size and adding negative sampling in combination with choosing the appropriate distribution for negative sampling, to the model are very important. Other hyper-parameters such as the window size, training time and minimum count do not make a difference when they are changed in isolation, though the combination of these hyper-parameters is somewhat important. We focused on studying the effect that negative sampling has when used in combination with other hyper-parameters. We found that optimising the parameters can produce a 31% performance improvement on the task. When comparing the code-switched model to a BOW baseline model and a pre-trained GloVe model, we found that the optimised Word2Vec model outperformed the pre-trained models and showed results that are competitive to the BOW model. These findings show both the importance of optimising certain hyper-parameters to fit the task, and the potential of embedding representations to process recognised multilingual speech.

6 Acknowledgements

We thank the Novus Group for providing the data used in this study. We also thank Dr. Charl van Heerden, CEO of Saigen for his contributions and providing the speech-recognition results of the data.

References

1. Barnett, R., Codó, E., Eppler, E., Forcadell, M., Gardner-Chloros, P., van Hout, R., Moyer, M., Torras, M.C., Turell, M.T., Sebba, M., Starren, M., Wensing, S.: The lides coding manual: A document for preparing and analyzing language interaction data version 1.1—july, 1999. International Journal of Bilingualism **4**(2), 131–132 (2000). https://doi.org/10.1177/13670069000040020101, https://doi.org/10.1177/13670069000040020101
2. Bojanowski, P., Celebi, O., Mikolov, T., Grave, E., Joulin, A.: Updating pre-trained word vectors and text classifiers using monolingual alignment (2019)
3. Boyd, K., Eng, K.H., Page, C.D.: Area under the precision-recall curve: Point estimates and confidence intervals. In: Blockeel, H., Kersting, K., Nijssen, S., Železný, F. (eds.) Machine Learning and Knowledge Discovery in Databases. pp. 451–466. Springer Berlin Heidelberg, Berlin, Heidelberg (2013)
4. Caselles-Dupré, H., Lesaint, F., Royo-Letelier, J.: Word2vec applied to recommendation: Hyperparameters matter. CoRR **abs/1804.04212** (2018), http://arxiv.org/abs/1804.04212
5. Chiu, B., Korhonen, A., Pyysalo, S.: Intrinsic evaluation of word vectors fails to predict extrinsic performance. In: Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP. pp. 1–6. Association for Computational Linguistics, Berlin, Germany (Aug 2016). https://doi.org/10.18653/v1/W16-2501, https://www.aclweb.org/anthology/W16-2501

6. Dillenberger, J.: Evaluation of model and hyperparameter choices in word2vec (2019)
7. Fanaeepour, M., Makarucha, A., Lau, J.H.: Evaluating word embedding hyperparameters for similarity and analogy tasks. CoRR **abs/1804.04211** (2018), `http://arxiv.org/abs/1804.04211`
8. Faruqui, M., Tsvetkov, Y., Rastogi, P., Dyer, C.: Problems with evaluation of word embeddings using word similarity tasks. In: Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP. pp. 30–35. Association for Computational Linguistics, Berlin, Germany (Aug 2016). https://doi.org/10.18653/v1/W16-2506, `https://www.aclweb.org/anthology/W16-2506`
9. Fellbaum, C.: WordNet: An Electronic Lexical Database. Bradford Books (1998)
10. Gambäck, B., Das, A.: Comparing the level of code-switching in corpora. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16). pp. 1850–1855. European Language Resources Association (ELRA), Portorož, Slovenia (May 2016), `https://www.aclweb.org/anthology/L16-1292`
11. Gladkova, A., Drozd, A.: Intrinsic evaluations of word embeddings: What can we do better? In: Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP. pp. 36–42. Association for Computational Linguistics, Berlin, Germany (Aug 2016). https://doi.org/10.18653/v1/W16-2507, `https://www.aclweb.org/anthology/W16-2507`
12. Guzman, G.A., Serigos, J., Bullock, B.E., Toribio, A.J.: Simple tools for exploring variation in code-switching for linguists. In: Proceedings of the Second Workshop on Computational Approaches to Code Switching. pp. 12–20. Association for Computational Linguistics, Austin, Texas (Nov 2016). https://doi.org/10.18653/v1/W16-5802, `https://www.aclweb.org/anthology/W16-5802`
13. Huang, C., Qiu, X., Huang, X.: Text classification with document embeddings. In: Sun, M., Liu, Y., Zhao, J. (eds.) Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data. pp. 131–140. Springer International Publishing, Cham (2014)
14. Jang, B., Inhwan, K., Kim, J.W.: Word2vec convolutional neural networks for classification of news articles and tweets. PLoS ONE (2019). https://doi.org/10.1371/journal.pone.0220976
15. Krebs, A., Paperno, D.: When hyperparameters help: Beneficial parameter combinations in distributional semantic models. In: Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics. pp. 97–101. Association for Computational Linguistics, Berlin, Germany (Aug 2016). https://doi.org/10.18653/v1/S16-2011, `https://www.aclweb.org/anthology/S16-2011`
16. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. CoRR **abs/1405.4053** (2014), `http://arxiv.org/abs/1405.4053`
17. Li, P., Liu, Y., Sun, M., Izuha, T., Zhang, D.: A neural reordering model for phrase-based translation. In: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers. pp. 1897–1907. Dublin City University and Association for Computational Linguistics, Dublin, Ireland (Aug 2014), `https://www.aclweb.org/anthology/C14-1179`
18. Linzen, T.: Issues in evaluating semantic spaces using word analogies. In: Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for

NLP. pp. 13–18. Association for Computational Linguistics, Berlin, Germany (Aug 2016). https://doi.org/10.18653/v1/W16-2503, https://www.aclweb.org/anthology/W16-2503

19. Mikolov, T., Chen, K., Corrado, G.S., Dean, J.: Efficient estimation of word representations in vector space. CoRR **abs/1301.3781** (2013)
20. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. CoRR **abs/1310.4546** (2013), http://arxiv.org/abs/1310.4546
21. Pratapa, A., Choudhury, M., Sitaram, S.: Word embeddings for code-mixed language processing. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 3067–3072. Association for Computational Linguistics, Brussels, Belgium (Oct-Nov 2018). https://doi.org/10.18653/v1/D18-1344, https://www.aclweb.org/anthology/D18-1344
22. Rijhwani, S., Sequiera, R., Choudhury, M., Bali, K., Maddila, C.S.: Estimating code-switching on Twitter with a novel generalized word-level language detection technique. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 1971–1982. Association for Computational Linguistics, Vancouver, Canada (Jul 2017). https://doi.org/10.18653/v1/P17-1180, https://www.aclweb.org/anthology/P17-1180
23. Schnabel, T., Labutov, I., Mimno, D., Joachims, T.: Evaluation methods for unsupervised word embeddings. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. pp. 298–307. Association for Computational Linguistics, Lisbon, Portugal (Sep 2015). https://doi.org/10.18653/v1/D15-1036, https://www.aclweb.org/anthology/D15-1036
24. Sinoara, R., Camacho-Collados, J., Rossi, R., Navigli, R., Rezende, S.: Knowledge-enhanced document embeddings for text classification. Knowledge-Based Systems (10 2018). https://doi.org/10.1016/j.knosys.2018.10.026

Analysing Coronavirus Trends in South Africa and Countries with a Similar Coronavirus Profile

Asad Jeewa[1][0000−0003−4329−8137], Edgar Jembere[1][0000−0003−1776−1925], Nirvana Pillay[1][0000−0003−4999−1215], Yuvika Singh[1][0000−0002−5430−3272], Serestina Viriri[1][0000−0002−2850−8645], and Anban Pillay[1][0000−0001−7160−6972]

School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, Westville 4000, South Africa
asad.jeewa@gmail.com, jemberee@ukzn.ac.za, nirvanap02@gmail.com, yuvikasingh@yandex.com, viriris@ukzn.ac.za, pillayw4@ukzn.ac.za

Abstract. South Africa has been hard hit by the novel Coronavirus (COVID-19) disease despite being one of the countries to adopt early use of non-pharmaceutical interventions to curb the spread of the disease. While these interventions seemed to have delayed the spread of the disease, the number of COVID-19 cases and fatalities still increased soon after the interventions were relaxed to levels even higher than some countries that did not introduce measures of similar severity. This raises questions as to why this is the case. This study therefore investigated the reasons why South Africa and countries with a similar coronavirus profile had such a trajectory in terms of the number of COVID-19 cases and fatalities. The study proposes a methodology for profiling and grouping countries with similar COVID-19 trajectories. This technique is termed as "$\alpha\sigma$-Nearest Neighbour". We used this method to find countries that have a COVID-19 profile similar to South Africa. The second step of the study involved analysing various socio-economic indicators and non-pharmaceutical interventions for the selected countries. Our results show that socio-economic factors are generally not significantly correlated with the spread and the fatalities of the virus, and thus they do not explain variability in the COVID-19 cases and fatalities. This was also found to be the case even for countries that have a similar COVID-19 profile to South Africa. Non-pharmaceutical interventions were found to largely explain variability in the trajectory the spread of the virus followed.

Keywords: Coronavirus · Clustering · Correlation

1 Introduction

The world has been challenged by an unprecedented pandemic, COVID-19, that has introduced uncertainty into daily lives, the economy and the future. The World Health Organisation (WHO) has declared COVID-19 a public health emergency of international concern [9] and countries around the world have been devastated. In such times, it is necessary to understand prevalence trends and infection spread so as to deduce the factors that contribute to the spread of the virus and the mitigation strategies that are most effective. This is important not only to inform public health policy as the pandemic still rages, but also to prepare for the next pandemic.

COVID-19 is a moderately infectious virus with a basic reproductive number (R_0) of 3.28; the R_0 is the average number of infections that result from an infected person in a fully susceptible population [9]. The virus is transmitted via the respiratory droplets of an infected person and may be transmitted through human-to-human contact or human-to-fomite contact. A fomite is any inanimate object that, when contaminated with or exposed to an infectious agent, can transfer disease to a new host. It is, therefore, important to consider both environmental and lifestyle conditions when determining infections trends. These conditions in turn are influenced by a myriad of socio-economic, public health and governmental factors. This makes the understanding and prediction of the spread of the virus an onerous task.

This work set out to determine the socio-economic factors that influence both the spread and mortality rate of the virus. We also examine the non-pharmaceutical interventions that have proven effective in controlling the spread and mortality of the virus.

To deal with the deluge of data, we developed a method to group countries based on their coronavirus disease profile which we term '$\alpha\sigma$-Nearest Neighbour". A feature vector is calculated for each country using the number of reported cases and deaths. The features are used to cluster countries based on the distance of the vector from a specified country, which is South Africa in our case. This approach was preferred to traditional clustering methods such as in [1] that did not produce satisfactory clusters. We then examined fifteen socio-economic indicators to determine which of these correlated most closely with the number of cases and number of deaths. This was done by computing the Kendall's tau-B correlation coefficient for both early and late thresholds. Bayesian analysis was then performed to associate a confidence coefficient with the correlations. Lastly, the non-pharmaceutical interventions (mitigation) factors were considered: The effectiveness of full lockdowns are highlighted for curbing the spread of the virus.

2 Literature Review

The global nature of the pandemic and the devastating effects of the virus has led to numerous studies with the primary focus on the epidemiology of the virus [22,19]. However, several studies have acknowledged the importance of socio-economic factors on the spread of the virus. High temperature, low humidity, lifestyle, co-morbidities, age, and personal hygiene further influenced the spread of the virus [16].

Most studies similar to this work have attempted to analyse a subset of countries. The usual approach has been to hand-pick the subset of countries. In this work, we attempted to identify similar countries automatically. A few studies have used similar techniques. For example, Alvarez et al. [1] implemented non-parametric, graph theoretic techniques to categorize countries based the number of daily infections over a period of at least a 100 days. One hundred and ninety-one countries were categorized into three clusters each exhibiting unique features. However, the clusters produced were very coarse and included such diverse countries as United States and Zimbabwe within the same cluster. A similar study was conducted by Chandu [3] in which 89 countries were categorized into two main clusters using K-means clustering. Similar work in [26] used hierarchical clustering to cluster countries according to active cases, active cases per population and active cases per area.

Several studies have examined socio-economic correlations. The study in [8] used bivariate correlation using Pearson's correlation index and demonstrated significant relationships between the incidence of the disease and GDP and flights per capita. Our work used the Kendall's tau-B correlation coefficient as the socio-economic variables, number of cases, and number of deaths were not approximately univariate normally distributed. A similar study reported in [2] was carried out in the very early stages of the pandemic. This study was focused on case fatality rates and concluded that mortality rates was correlated with factors that suggest developing countries with poor economies. The study in [15] used Spearman correlation and Principal Component Analysis on a handpicked set of eight countries and also found a correlation with GDP.

Mitigation strategies have been an essential component of any response to the pandemic. These strategies have included regulations to limit person-to-person contact by means of quarantine and social distancing as well as isolating infected people [9]. The work in [4] analysed the effect of mitigation but only four hand-picked countries were studied. Their work examined neural networks to predict the effect of mitigation strategies. A study in [11] performed an extensive analysis of the non-pharmaceutical interventions in 11 European countries. The work analysed the effect of the interventions on the total number of cases and deaths of the countries, two metrics commonly used to represent the profile of a country. The work highlighted the difficulties in analysing single interventions. Also, since multiple interventions are implemented at the same time, it is difficult to disentangle their effects. It was also observed that deaths were under-reported in countries and due to limited testing, the actual number of cases is much higher than those reported. The authors also constructed a mathematical model that uses historical data of previous outbreaks to pre-empt future ones. Similar work is seen in [16,17].

3 Methodology

The data analysis was done in three stages: First, $\alpha\sigma$-nearest neighbour was used to find countries that had similar disease trajectories as South Africa. The identified countries were then studied to determine if the disease characteristics correlated with socio-economic factors. The last stage analysed the effect of non-pharmaceutical interventions on the spread of the disease.

3.1 $\alpha\sigma$-Nearest Neighbour

The goal of this analysis was to establish a set of countries with a similar "coronavirus profile" to South Africa's. The first step was to select features to represent this profile. *New cases per million* and *new deaths per million* were selected: they are more suitable than using the total cases and total deaths per million as it reflects the current landscape of cases and mortality. The *total* figures are constantly increasing so in order to use them, one would have to analyse the entire time-series or investigate the rate of change. The "per million" metric was chosen since using absolute figures is misleading and depicts a skewed picture of the cases and deaths in a country.

We wished to analyse coronavirus trends by assessing how the number of cases and deaths developed over time. However, it is not feasible to sample daily figures for each metric, due to dimensionality concerns. We alleviated this by sampling the total cases and deaths in fixed intervals i and then calculating the new case and deaths within i days. If there were no reported figures for a specific day, the closest date is found from the dataset. If the divergence is beyond a fixed threshold of $i/3$, the country i is discarded from the dataset since there is not enough data for the country. This process results in an abstracted view of a countries' coronavirus profile without losing the salient information. The start day s and end day e are also specified. s refers to the number of days since the first case or death was reported. For this work, we fixed s and i at 15 and e at 150. This resulted in a large enough period for analysing coronavirus trends across the different countries. Sampling using a fixed day interval also allowed us to compare every country independently, as opposed to using dates. A feature vector was defined in this manner for every country in the dataset by repeating the process for every selected coronavirus feature.

Various other metrics, such as the recovery rate and number of tests preformed, were considered. Testing statistics indicate the divergence between the number of cases reported and the actual number of cases in a country. Two countries may seem to have a similar number of cases but if one country is testing on a smaller scale, this could mean that there are actually significantly more cases. However access to such data is very limited since many countries do not make these statistics publicly available. When we incorporated these metrics into the methodology, more than half the countries were excluded from the comparison. Feature selection was therefore restricted based on data availability and reliability. Other factors, such as human behaviour, is very difficult to quantify, while factors such as the number of hospital beds and the positive rate were not generally available.

There are also various ways of grouping or clustering the countries. A concern with clustering techniques is that is is difficult to decide on the number of clusters: some algorithms require this to be specified beforehand, while others like affinity propagation [7] calculate this variable. Hyperparameter tuning takes on extra importance and is used to control the number of clusters. However, the issue with clustering is that it aims to maximise inter-cluster distance and minimise intra-cluster distance. This joint objective function means that two countries that are similar may well be grouped in different clusters.

We chose a simple but effective approach: find the euclidean distance from the feature vector of South Africa to all other feature vectors in the dataset. This list was then sorted to define the closest countries. Instead of selecting a fixed number of countries, we define a cut-off point of 2 standard deviations and select only the closest n countries in this manner. For this reason, we term this approach as "$\alpha\sigma$-Nearest Neighbour".

Special care was taken to avoid single features dominating. The number of cases is naturally higher than the number of deaths, sometimes by a large factor. It was therefore important to ensure that both the cases and the deaths contributed equally to the grouping. Z-Score Normalisation was used for this purpose. To validate the results, the countries were also grouped based on cases and deaths separately to asses whether any countries were very similar in one aspect but different in the other.

The last step involves discarding any countries with a small population size. This is because the "per million" metric does not take into account the population size. The population size also heavily influences the choice and implementation of non-pharmaceutical interventions. We defined a fixed threshold of 10 million, which translates to approximately 17% of the population of South Africa.

3.2 Correlation Analysis

The next step in the analysis was an investigation of correlations between the total number of cases and total number of deaths with socio-economic factors. We obtained multiple datasets for fifteen different socio economic factors, as seen in Table 1 and Table 2 and performed cross-sectional correlation with the cases and deaths of a country at fixed days. We computed the correlation between the total number of cases and socio-economic factors at (i) 30 and (ii) 150 days after the first reported case. The same was done for the total number of deaths, but at (i) 45 and (ii) 150 days after the first death was reported.

Initial correlation analysis of the socio-economic variables with both total number of cases and total number of deaths on all the countries showed very weak and anecdotal correlations. We therefore performed correlation analysis within the group of countries that were identified using the methodology defined in subsection 3.1. In order to get enough data for more reliable statistical analysis, we increased the number of countries for our correlation analysis by including countries within a distance of 2.5 standard deviation from South Africa. We also analysed the variance of socio economic features. Outliers were removed from the data before computing the correlations. The Inter-Quartile Range method [25] was used to remove outliers i.e. countries with a socio-economic feature beyond $3 * IQR$ were excluded from the analysis. This is slightly larger than the traditional multiplier of 1.5 since we wanted to remove only extreme outliers that would heavily influence the correlation. The validity of the method was verified by visualising the data. South Africa, for example, was shown to have a significantly higher unemployment rate and lower life expectancy than other countries in the grouping.

The Pearson correlation coefficient is commonly used with coronavirus data [2,8]. However, due to the socio-economic variables, number of cases, and number of deaths not being approximately univariate normally distributed, the Kendall's Tau-B [6] correlation coefficient was used instead. For every socio-economic feature, we computed the Kendall's Tau-B Correlation Coefficient for both the early and late thresholds. Bayesian Correlation Analysis [24] was performed on the data to allow us to associate a confidence coefficient. Non-informative priors [18] that assume a uniform distribution of the correlation coefficients was used for the analysis. The JASP package [1] was used for analysis.

This analysis was meant to investigate the reasons for the similarity in Coronavirus profiles of the grouped countries. We postulated that the similarity could be influenced by socio-economic similarities among the countries or it could be a direct result of similar non-pharmaceutical interventions that were employed by the countries.

It is important to point out that there are various other factors that influence the spread of the virus. It is difficult to quantify human behaviour and the success of a specific mitigation strategy depends directly on how effectively the citizens of the country adhere to the restrictions.

3.3 Mitigation Analysis

It was not possible to compare all the countries of the world directly in terms of the non-pharmaceutical interventions made across the world. This study was restricted to countries clustered as described above. This allowed an in-depth study and comparison of the mitigation strategies to identify patterns and trends.

While data on non-pharmaceutical interventions was available [2], the variance in the nature of the mitigation strategies between countries made comparison difficult. For example, a lockdown could be regional or national and the conditions of the lockdown may be different as well. Thus we focused only on lockdown measures. The initial implementation of a full lockdown and the first date of easing of the measure were plotted on the case/death graph and a qualitative analysis was performed.

We also compared the mitigation strategies of South Africa to European countries that were affected by the virus in the very early stages of the global pandemic. We used the methodology defined in subsection 3.1 to find countries that match South Africa's coronavirus profile in the first 30 days of the local pandemic. This was done to analyse the effect of the non-pharmaceutical interventions in countries that diverged from South Africa's curve. It should be noted that the timeline of the disease heavily influences how countries mitigate: countries that saw a later spike in the cases were able to observe the successes and failures of others and therefore adapt accordingly.

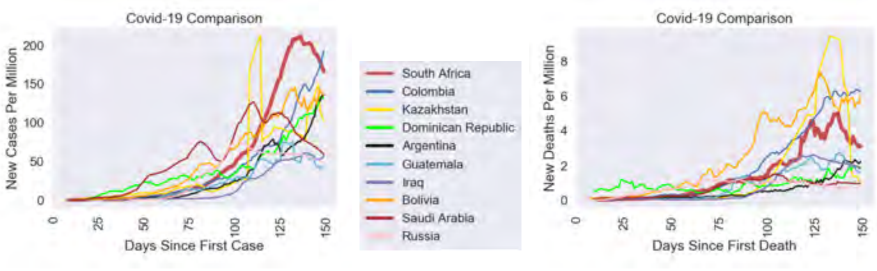
[1] https://jasp-stats.org/

[2] http://www.acaps.org

4 Results

4.1 $\alpha\sigma$-Nearest Neighbour

Using the method detailed in subsection 3.1, the following nine countries were found to have the most similar coronavirus profile to South Africa: Colombia, Kazakhstan, Dominican Republic, Argentina, Guatemala, Iraq, Bolivia, Saudi Arabia and Russia. The new cases per million and new deaths per million graphs are depicted with a rolling average of seven. It shows that all the countries have cases and deaths within a similar range.



(a) Cases per million

(b) Deaths per million

Fig. 1: Coronavirus profiles of the grouped countries

4.2 Correlation Analysis

After increasing the distance from South Africa to 2.5 standard deviations, the following additional countries were added for this step: Iran, Romania, Brazil, Egypt, Ukraine, Ecuador, Bangladesh, Mexico, Pakistan and India. The main results from Bayesian correlation analysis are shown in Table 1 and Table 2.

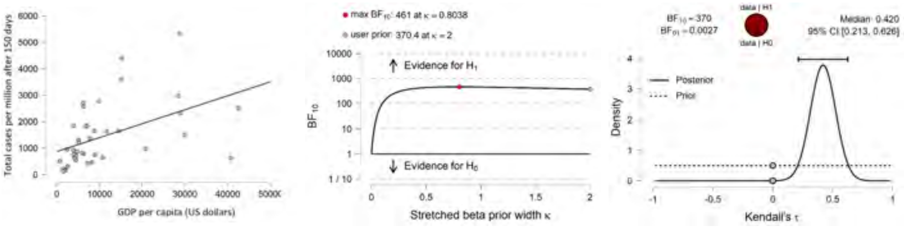
Table 1: Correlation between various socio-economic variables and the total number of cases

Socio-economic variable	Number of days	Number Countries	Kendall's Tau-b	BF_{10}	Lower 95% CI	Upper 95% CI
Unemployment rate	30	41	0.198	0.692	-0.014	0.394
	150	41	0.091	0.185	-0.118	0.291
Sex ratio	30	37	-0.267	2.064	-0.470	-0.041
	150	37	0.267	2.064	-0.470	-0.041
Tourist/visitor arrivals	30	36	-0.159	0.352	-0.369	0.066
	150	36	-0.187	0.507	-0.397	0.039

Continued on next page

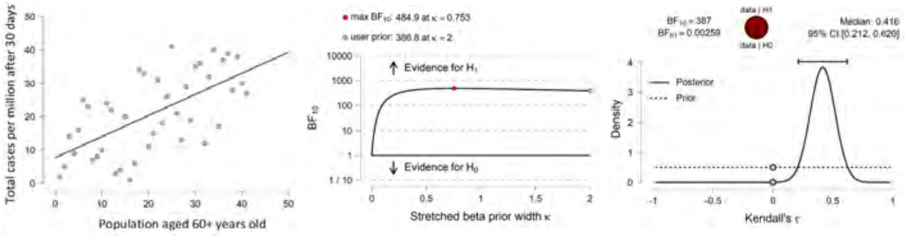
Table 1 – *Continued from previous page*

Socio-economic variable	Number of days	Number Countries	Kendall's Tau-b	BF_{10}	Lower 95% CI	Upper 95% CI
Population Density	30	40	0.028	0.137	-0.180	0.234
	150	40	-0.162	0.388	-0.362	0.052
Population aged 60+ years old	30	41	0.434	**386.77**	0.212	0.620
	150	41	0.139	0.296	-0.071	0.337
People with basic handwashing facilities	30	14	0.143	0.426	-0.219	0.450
	150	14	-0.231	0.620	-0.521	0.147
People using basic sanitation services	30	39	0.137	0.431	-0.079	0.333
	150	39	0.148	0.486	-0.069	0.343
Mortality rate attributed to unsafe sanitation	30	33	-0.218	0.714	-0.435	0.020
	150	33	0.067	0.260	-0.162	0.285
Literacy rate, adult total	30	14	-0.231	0.432	-0.554	0.144
	150	14	0.143	0.288	0.223	0.476
Life expectancy at birth	30	41	0.395	**98.027**	0.175	0.583
	150	41	0.324	**11.321**	0.107	0.515
Intentional homicide rates	30	27	-0.029	0.164	-0.279	0.225
	150	27	0.302	1.861	0.032	0.535
Health personnel: Physicians	30	12	0.333	0.753	-0.667	0.087
	150	12	-0.061	0.375	-0.407	0.312
GDP per capita	30	40	0.303	5.790	0.083	0.496
	150	40	0.438	**370.41**	0.213	0.626
Exchange rates to US Dollar	30	33	-0.217	0.705	-0.434	0.021
	150	33	-0.354	9.705	-0.109	-0.564
Government health expenditure	30	36	0.368	**20.522**	0.133	0.569
	150	36	0.279	2.465	0.049	0.484



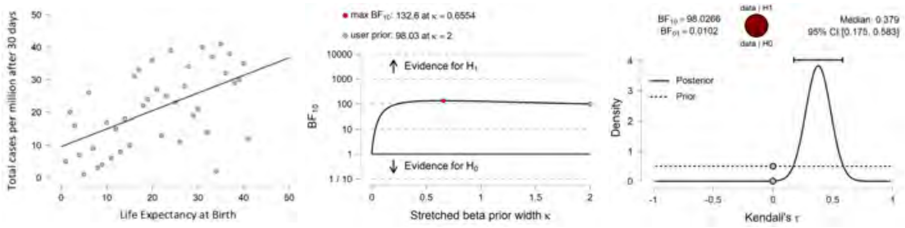
(a) Scatter Plot (b) Robustness analysis (c) Density analysis

Fig. 2: Bayesian analysis for GDP per capita (dollars) and total cases per million after 150 days



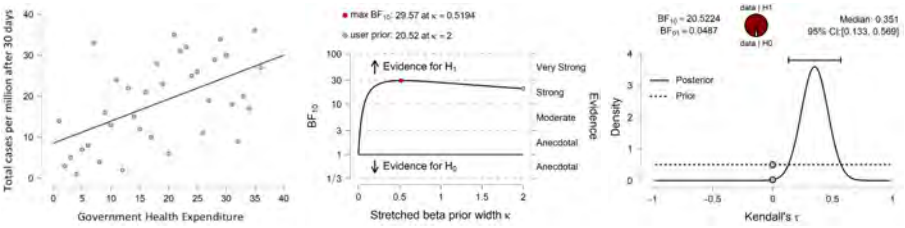
(a) Scatter Plot (b) Robustness analysis (c) Density analysis

Fig. 3: Bayesian analysis for population aged 60+ years old (percentage) and total cases per million after 30 days



(a) Scatter Plot (b) Robustness analysis (c) Density analysis

Fig. 4: Bayesian analysis for life expectancy at birth and total cases per million after 30 days



(a) Scatter Plot (b) Robustness analysis (c) Density analysis

Fig. 5: Bayesian analysis for government health expenditure and total cases per million after 30 days

The results show weak and anecdotal correlations for most of the socio-economic variables. When correlating socio-economic variables and the number of cases 150 days after the first case, moderate correlations were observed for *life expectancy at birth* (0.324), *international homicide rates* (0.302), *GDP per capita* (0.438), and *exchange rates to US Dollar*(-0.354). The correlation on *international homicide rates* was found to be anecdotal with Bayes Factors BF_{10} of 1.861 in favour of correlation. The correlation on *exchange rates to US Dollar* was moderately supported with a BF_{10} of 9.705 indicating a moderately decisive likelihood ("evidence") of the correlation. The correlation on *GDP per capita* had a strongly decisive likelihood with a BF_{10} of 370.41, and the robustness analysis showed that the correlation is insensitive to changes in the prior width (see Figure 2).

The correlations with the total number of cases 30 days after the first case were moderate for *population aged 60+ years* (0.434), *life expectancy at birth* (0.395), *health personnel: physicians* (-0.333), and *government health expenditure* (0.368). The correlation on *health personnel: physicians* was found to be insignificant with a BF_{10} of 0.753, which favours no correlation. The correlations on *population aged 60+ years*, *life expectancy at birth*, and *government health expenditure* were decisively supported by the data with BF_{10} values 388.77, 98.027, and 20.522 respectively. Figure 3, Figure 4, and Figure 5 show the results of the robustness analysis of the supposed correlation with *population aged 60+ years*, *life expectancy at birth*, and *government health expenditure*. The figures show that, except for very small prior widths, the Bayes factors are relatively stable confirming the robustness of the analysis for all the three socio-economic factors.

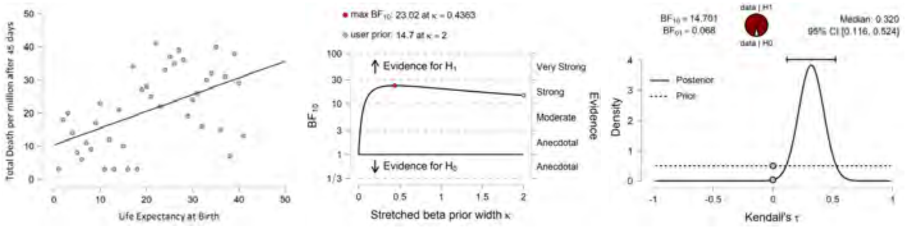
Table 2: Correlation between various socio-economic variables and the total number of deaths

Socio-economic variable	Number of days	Number Countries	Kendall's Tau-b	BF_{10}	Lower 95% CI	Upper 95% CI
Unemployment rate	45	41	-0.079	0.170	-0.280	0.128
	150	41	0.230	1.233	0.017	0.425
Sex ratio	45	37	-0.206	0.684	-0.412	0.018
	150	37	-0.294	3.671	-0.496	-0.067
Tourist/Visitor arrivals	45	36	0.179	0.454	-0.047	0.388
	150	36	-0.238	1.123	-0.445	-0.010
Population Density	45	40	0.170	0.433	-0.045	0.369
	150	40	-0.179	0.500	-0.379	0.035
Population aged 60+ years old	30	41	0.267	2.714	0.052	0.460
	150	41	0.217	0.963	0.004	0.412
People with basic handwashing facilities	45	16	-0.117	0.255	-0.432	0.221
	150	16	-0.100	0.242	-0.417	0.236

Continued on next page

Table 2 – *Continued from previous page*

Socio-economic variable	Number of days	Number Countries	Kendall's Tau-b	BF_{10}	Lower 95% CI	Upper 95% CI
People using at least basic sanitation services	45	39	0.007	0.134	-0.203	0.216
	150	39	-0.137	0.431	-0.334	0.079
Mortality rate attributed to unsafe sanitation	45	33	-0.105	0.211	-0.328	0.127
	150	33	0.274	1.779	0.032	0.487
Literacy rate, adult total	30	14	0.143	0.288	-0.223	0.476
	150	14	-0.231	0.432	-0.554	0.144
Life expectancy at birth	45	41	0.334	**14.701**	0.116	0.524
	150	41	0.202	0.742	-0.010	0.398
Intentional homicide rates	45	27	-0.089	0.198	-0.335	0.168
	150	27	0.342	3.708	0.070	0.573
Health personnel: Physicians	45	12	0.061	0.250	-0.325	0.430
	150	12	0.295	0.587	-0.121	0.633
GDP per capita	45	40	0.237	1.341	0.020	0.433
	150	40	0.272	2.791	0.054	0.467
Exchange rates to US Dollar	45	33	-0.108	0.214	-0.330	0.125
	150	33	-0.099	0.202	-0.322	0.133
Government health expenditure	45	36	0.188	0.513	-0.039	0.397
	150	36	0.324	**6.613**	0.091	0.526



(a) Scatter Plot (b) Robustness analysis (c) Density analysis

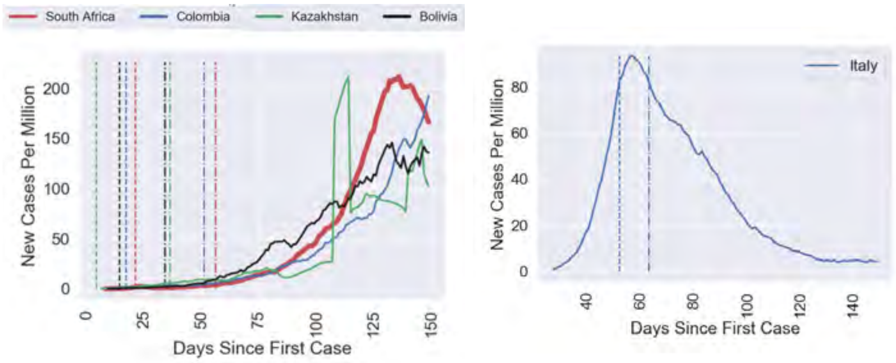
Fig. 6: Bayesian analysis for life expectancy at birth and total deaths per million after 45 days

When correlating the socio-economic variables and the number of deaths 150 days after the first case, moderate correlations were observed for Intentional homicide rates (0.342) and government health expenditure (0.324). These correlations were moderately decisively supported by the data with BF_{10} values of 3.708 and 6.613 respectively. The correlations with the total number of deaths 45 days after the first case were moderate for life expectancy at birth (0.334). This correlation was also found to be decisively supported by the data with a BF_{10} value of 14.701. The robustness analysis showed that the BF_{10} value was relatively stable for prior values in the range from 0 to 2 (see Figure 6), confirming robustness of the analysis.

The correlations of the variables GDP per capita, and government health expenditure, with the number of cases/deaths were often counter-intuitive. One possible reason for this observation is that there are various factors at play including un-quantifiable variables that influence the spread of the virus, like human behaviour.

4.3 Mitigation Analysis

As discussed previously, this work focused only on the lockdown as a mitigation strategy. A full or partial lockdown involves severe restrictions on freedom of movement and may include curfews and restrictions on business openings. In general, a full lockdown is shown to be effective in curbing the rate of the spreads of the virus i.e. "flattening the curve". We depict the point at which the lockdown was stared with dashed lines and the point at which restrictions began to be eased with dashed-dotted lines.



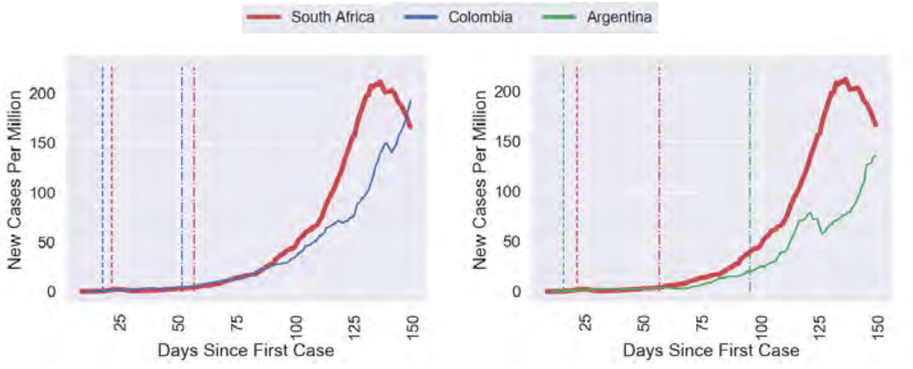
(a) Early lockdown

(b) Lockdown at the virus peak

Fig. 7: Examples of different lockdown strategies

Figure 7a shows examples of countries that introduced a full lockdown in the early stages of the local pandemic when cases were very low. The number of new cases reported remained low during the lockdown but began to rise drastically when it was lifted. This indicates that these countries were able to give themselves time to prepare for the peak but it is not possible to avoid it entirely due to the nature of the virus and the lack of immunity. We theorise that this is one of the factors that contributed towards South Africa's mortality rate being lower.

Figure 7b shows the alternate approach where countries introduced a full lockdown only once cases began to rise sharply, as opposed to acting pre-emptively. This causes the cases to drop immediately thereafter. This approach is often observed in countries that were affected by the virus in the early stages of the global pandemic. Italy, for example, implemented a short lockdown only as they approached the "peak" of the virus.



(a) Cases per million

(b) Deaths per million

Fig. 8: A comparison of the full lockdown observed in South Africa and Colombia

Argentina (see Figure 8a) experienced a longer lockdown which likely contributed to fewer cases and deaths. However, the length of the lockdown has to be balanced with the economic devastation that they cause [13].

Colombia in Figure 8b shows a similar lockdown duration. However, Colombia has more cases. This indicates that while lockdowns are one of the most effective means of curbing the spread of the virus, there are various other factors. It is often a combination of multiple interventions instead of a single intervention that allows a country to manage the spread of the virus.

5 Conclusions and Future Work

This paper proposed a methodology for grouping countries based on the COVID-19 profile and analysed the correlations between a set of socio-economic variables with the number of coronavirus cases deaths. We also investigated the effect of lockdowns on the number of cases and deaths.

The results highlight the effectiveness of our methodology in finding countries that have a similar trend in terms of both the spread of the virus and mortality rates. An analysis of the correlations of socio-economic variables with the coronavirus profile variables did not show any strong correlation within the group. Significant moderate correlations with the total number of cases 30 days after the first case was reported were observed for *population aged 60+ years*, *life expectancy at birth*, and *government health expenditure. Life expectancy at birth* and *GDP per capita* were found to be significantly correlated with the total number of cases 150 days after the first case was reported. Significant moderate correlations were observed between *life expectancy at birth* and the total number of deaths, 45 days after the first reported death.

The analysis of the effect of the mitigation strategies shows that the countries with a similar coronavirus profile to South Africa employed an early full lockdown. This coupled with the results from the correlation analysis seems to suggest that the coronavirus profiles were heavily influenced by the non-pharmaceutical interventions employed, rather than the socio-economic statuses of the countries. This is further supported by the fact that the correlation between *population aged 60+ years* (a well known corona virus risk factor) and the total number of cases is significant when considering the number of cases, 30 days after the first reported case, but the correlation does not exist when we consider the number of cases after 150 days. This is likely due to the interventions that the counties employed to protect their senior citizens. However, further analysis is necessary to ascertain whether the non-pharmaceutical interventions geared towards protecting the aged population resulted in a smaller number of senior citizens being infected by the virus.

There are various directions for future work. We would like to introduce additional features to the $\alpha\sigma$-Nearest Neighbour algorithm in subsection 3.1, especially testing information. We also wish to perform correlation analysis on more variables such as environmental features like the average temperature and humidity. It is also necessary to investigate the effect of other mitigation factors, beyond lockdowns.

References

1. Alvarez, E., Brida, J.G., Limas, E.: Comparisons of COVID-19 dynamics in the different countries of the World using Time-Series clustering. medRxiv p. 2020.08.18.20177261 (Aug 2020). https://doi.org/10.1101/2020.08.18.20177261, https://www.medrxiv.org/content/10.1101/2020.08.18.20177261v1, publisher: Cold Spring Harbor Laboratory Press
2. Asfahan, S., Shahul, A., Chawla, G., Dutt, N., Niwas, R., Gupta, N.: Early trends of socio-economic and health indicators influencing case fatality rate of COVID-19 pandemic. Monaldi Archives for Chest Disease **90**(3) (Jul 2020). https://doi.org/10.4081/monaldi.2020.1388, https://www.monaldi-archives.org/index.php/macd/article/view/1388
3. Chandu, V.: Identification of spatial variations in covid-19 epidemiological data using k-means clustering algorithm: a global perspective. medRxiv (2020). https://doi.org/10.1101/2020.06.03.20121194, https://www.medrxiv.org/content/early/2020/06/05/2020.06.03.20121194
4. Dandekar, R., Barbastathis, G.: Quantifying the effect of quarantine control in Covid-19 infectious spread using machine learning. preprint, Epidemiology (Apr

2020). https://doi.org/10.1101/2020.04.03.20052084, http://medrxiv.org/lookup/doi/10.1101/2020.04.03.20052084

5. Dietz, L., Horve, P.F., Coil, D.A., Fretz, M., Eisen, J.A., Van Den Wymelenberg, K.: 2019 novel coronavirus (covid-19) pandemic: Built environment considerations to reduce transmission. mSystems **5**(2) (2020). https://doi.org/10.1128/mSystems.00245-20, https://msystems.asm.org/content/5/2/e00245-20
6. van Doorn, J., Ly, A., Marsman, M., Wagenmakers, E.J.: Bayesian inference for kendall's rank correlation coefficient. The American Statistician **72**(4), 303–308 (2018). https://doi.org/10.1080/00031305.2016.1264998, https://doi.org/10.1080/00031305.2016.1264998
7. Dueck, D.: Affinity propagation: Clustering data by passing messages. PhD Thesis. University of Toronto (2009)
8. Gangemi, S., Billeci, L., Tonacci, A.: Rich at risk: socio-economic drivers of COVID-19 pandemic spread. Clinical and Molecular Allergy **18**(1), 12 (Jul 2020). https://doi.org/10.1186/s12948-020-00127-4, https://doi.org/10.1186/s12948-020-00127-4
9. Harapan, H., Itoh, N., Yufika, A., Winardi, W., Keam, S., Te, H., Megawati, D., Hayati, Z., Wagner, A.L., Mudatsir, M.: Coronavirus disease 2019 (covid-19): A literature review. Journal of Infection and Public Health **13**(5), 667 – 673 (2020). https://doi.org/https://doi.org/10.1016/j.jiph.2020.03.019, http://www.sciencedirect.com/science/article/pii/S1876034120304329
10. Higgins, T.S., Wu, A.W., Sharma, D., Illing, E.A., Rubel, K., Ting, J.Y., Snot Force Alliance: Correlations of Online Search Engine Trends With Coronavirus Disease (COVID-19) Incidence: Infodemiology Study. JMIR Public Health and Surveillance **6**(2), e19702 (May 2020). https://doi.org/10.2196/19702, https://publichealth.jmir.org/2020/2/e19702
11. Imperial College COVID-19 Response Team, Flaxman, S., Mishra, S., Gandy, A., Unwin, H.J.T., Mellan, T.A., Coupland, H., Whittaker, C., Zhu, H., Berah, T., Eaton, J.W., Monod, M., Ghani, A.C., Donnelly, C.A., Riley, S.M., Vollmer, M.A.C., Ferguson, N.M., Okell, L.C., Bhatt, S.: Estimating the effects of non-pharmaceutical interventions on COVID-19 in Europe. Nature (Jun 2020). https://doi.org/10.1038/s41586-020-2405-7, http://www.nature.com/articles/s41586-020-2405-7
12. Jang, S., Han, S.H., Rhee, J.Y.: Cluster of coronavirus disease associated with fitness dance classes, south korea. Emerging infectious diseases **26**(8), 1917 (2020)
13. Khailaie, S., Mitra, T., Bandyopadhyay, A., Schips, M., Mascheroni, P., Vanella, P., Lange, B., Binder, S., Meyer-Hermann, M.: Estimate of the development of the epidemic reproduction number rt from coronavirus sars-cov-2 case data and implications for political measures based on prognostics. medRxiv (2020)
14. Klingwort, J., Schnell, R.: Critical Limitations of Digital Epidemiology. Survey Research Methods pp. 95–101 Pages (Jun 2020). https://doi.org/10.18148/SRM/2020.V14I2.7726, https://ojs.ub.uni-konstanz.de/srm/article/view/7726, artwork Size: 95-101 Pages Publisher: Survey Research Methods
15. Kumar, A., Rani, P., Kumar, R., Sharma, V., Purohit, S.R.: Data-driven modelling and prediction of COVID-19 infection in India and correlation analysis of the virus transmission with socio-economic factors. Diabetes & Metabolic Syndrome: Clinical Research & Reviews **14**(5), 1231–1240 (Sep 2020). https://doi.org/10.1016/j.dsx.2020.07.008, http://www.sciencedirect.com/science/article/pii/S187140212030254X

16. Lakshmi Priyadarsini, S., Suresh, M.: Factors influencing the epidemiological characteristics of pandemic COVID 19: A TISM approach. International Journal of Healthcare Management **13**(2), 89–98 (Apr 2020). https://doi.org/10.1080/20479700.2020.1755804, https://www.tandfonline.com/doi/full/10.1080/20479700.2020.1755804
17. Liu, Y., Gu, Z., Xia, S., Shi, B., Zhou, X.N., Shi, Y., Liu, J.: What are the underlying transmission patterns of COVID-19 outbreak? An age-specific social contact characterization. EClinicalMedicine **22**, 100354 (May 2020). https://doi.org/10.1016/j.eclinm.2020.100354, https://linkinghub.elsevier.com/retrieve/pii/S2589537020300985
18. Martz, H.F.: Bayesian Reliability Analysis. American Cancer Society (2008). https://doi.org/10.1002/9780470061572.eqr081, https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470061572.eqr081
19. Pradhan, D., Biswasroy, P., Kumar Naik, P., Ghosh, G., Rath, G.: A Review of Current Interventions for COVID-19 Prevention. Archives of Medical Research **51**(5), 363–374 (Jul 2020). https://doi.org/10.1016/j.arcmed.2020.04.020, https://linkinghub.elsevier.com/retrieve/pii/S0188440920306159
20. Rajesh, R.: Technological capabilities and supply chain resilience of firms: A relational analysis using Total Interpretive Structural Modeling (TISM). Technological Forecasting and Social Change **118**, 161–169 (May 2017). https://doi.org/10.1016/j.techfore.2017.02.017, http://www.sciencedirect.com/science/article/pii/S004016251730197X
21. Salathe, M., Bengtsson, L., Bodnar, T.J., Brewer, D.D., Brownstein, J.S., Buckee, C., Campbell, E.M., Cattuto, C., Khandelwal, S., Mabry, P.L., et al.: Digital epidemiology. PLoS Comput Biol **8**(7), e1002616 (2012)
22. Shereen, M.A., Khan, S., Kazmi, A., Bashir, N., Siddique, R.: COVID-19 infection: Origin, transmission, and characteristics of human coronaviruses. Journal of Advanced Research **24**, 91–98 (Jul 2020). https://doi.org/10.1016/j.jare.2020.03.005, https://linkinghub.elsevier.com/retrieve/pii/S2090123220300540
23. Sornette, D., Mearns, E., Schatz, M., Wu, K., Darcet, D.: Interpreting, analysing and modelling COVID-19 mortality data. Nonlinear Dynamics **101**(3), 1751–1776 (Aug 2020). https://doi.org/10.1007/s11071-020-05966-z, https://doi.org/10.1007/s11071-020-05966-z
24. Sánchez-Taltavull, D., Ramachandran, P., Lau, N., Perkins, T.: Bayesian correlation analysis for sequence count data. **11**(10) (Oct 2016)
25. Yang, J., Rahardja, S., Fränti, P.: Outlier detection: How to threshold outlier scores? In: Proceedings of the International Conference on Artificial Intelligence, Information Processing and Cloud Computing. AIIPCC '19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3371425.3371427, https://doi.org/10.1145/3371425.3371427
26. Zarikas, V., Poulopoulos, S.G., Gareiou, Z., Zervas, E.: Clustering analysis of countries using the COVID-19 cases dataset. Data in Brief **31**, 105787 (Aug 2020). https://doi.org/10.1016/j.dib.2020.105787, http://www.sciencedirect.com/science/article/pii/S2352340920306818

Probabilistic Distributional Semantic Methods for Small Unlabelled Text

Jocelyn Mazarura[1] $^{[0000-0003-4598-0834]}$, Alta de Waal[1,2] $^{[0000-0001-8121-6249]}$, and Tristan Harris[1] $^{[0000-0002-4848-760X]}$

[1] Department of Statistics, University of Pretoria, Pretoria, South Africa
jocelyn.mazarura@up.ac.za, alta.dewaal@up.ac.za, th240498@gmail.com
[2] Centre for Artificial Intelligence (CAIR), Pretoria, South Africa

Abstract. Distributional semantic models (DSMs) explore the meaning in language and aim to create semantic representations through learning by association. They are based on the assumption that the meaning of a word can be inferred from its usage in combination with other words. Its applications range from community question answering systems to word sense disambiguation. A vector-space approach is the most common DSM methodology and a very recent and successful example is word embeddings such as word2vec. Once words are represented in Euclidean space, the applications are almost endless, ranging from expanding search requests to sentiment analysis and semantic similarity. An alternative to viewing the aggregate of co-occurrence counts for a word as constituting a vector is viewing it as a probability distribution. Such a probabilistic framework has many advantages: by means of latent variable models, not only dimensionality reduction, but also clustering can be achieved. Furthermore, a Bayesian approach to parameter distributions facilitates regularization and incorporates prior information. What we are most interested in, however, is a methodology's ability to handle uncertainty in small corpora. By 'small corpora' we mean both small in terms of the number of documents and small in terms of the length of documents. In this work, we present a special case of a small, unstructured corpus and compare the performance of two probabilistic DSMs, namely Latent Dirichlet Allocation (LDA) and Gamma-Poisson Mixture (GPM). We expand the investigation to short text, which is even more sparse in word co-occurence statistics. The documents in this dataset are unlabelled and we illustrate the additional benefit of probabilistic DSMs of labelling documents. Finally, we introduce a metric, relevance index. Relevance index translates the similarity distance between a corpus of interest and a test document to the probability of the test document to be semantically similary to the corpus of interest.

Keywords: distributional semantics · topic models · short text.

1 Introduction

Consider the scenario where a company, e.g. Discovery Ltd., wants to gather information on what the public thinks about their services. This is a typical task for the resident data scientist who is then tasked with scouring digital media for documents including the keyword 'Discovery'. Digital media includes social media posts, digital newspapers and blogs. Keyword search methods are quite straightforward and output all documents in the search space containing the keyword. The next step is to determine if all these documents actually relate to the intended keyword. In the case of our Discovery example, some documents could refer to the Discovery Channel or discovery with regards to exploration in the wilderness, etc. The question driving this research is:

Can we determine if a new text is semantically related to a corpus of interest based on unstructured information only?

The problem is further defined by the following challenges: Let us refer to the corpus of interest as the reference corpus. For our Discovery example, this is a set of documents that we *know* to have the correct context we are interested in. The solution to this problem is to transform the corpus into a lower dimensional representation in accordance with the manifold hypothesis, which states that higher dimensional data lies within lower dimensional forms embedded inside higher-dimensional spaces [3].

This is where the field of distributional semantics comes into play. Distributional semantic methods tackle problems of meaning [12]. They are distributional in the sense that their parameters are learned through context from other observed co-occuring words. Distributional semantic methods transform high dimensional text into a lower dimensional vector space which is amenable to linear algebra computations. From here on the natural language processing (NLP) applications are almost endless, ranging from collaborative filtering [4, 14], aspect-based sentiment analysis [2] and text classification [9]. Under the vector space approach, the acquired transformation $\mathbf{f}_w$ is defined as a vector for $w \in \mathcal{C}$, where w is each word in the corpus vocabulary $\mathcal{C}$. These are called vector space models (VSM) and they are generally based on a co-occurrence matrix like a term-document matrix of the corpus. Furthermore, the distributional hypothesis states that words with similar $\mathbf{f}_w$ vectors have similar meaning [6]. An example of a VSM is the tf-idf (term frequency-inverse document frequency) model where words are defined by a simple function of the frequency of its nearest neighbours. However, this method generates very long, sparse vectors. In contrast, Word2vec can create short and dense vectors that have useful semantic properties. The comparison between two vectors is made with similarity functions such as cosine similarity [11]. With recent advances in deep learning, word embeddings such as *word2vec* [10] became popular distributional representations of words to such an extent that pre-trained word embeddings have been developed and are available for open-source use [3]. These pre-trained embeddings are trained on millions if not billions of words from web-based corpora [13].

If the vector $\mathbf{f}_w$ is normalised to unit sum, then it parameterises a discrete distribution which can be defined as the conditional probability of observing a particular context given that we observed w [12]. We don't need to search far

[3] `https://nlp.stanford.edu/projects/glove/`

for such a probabilistic representation, as topic models such as the well-known Latent Dirichlet Allocation (LDA) [1] and the recent Gamma Poisson mixture (GPM) [8] for short text, produce such representations. They decompose high dimensional count vectors into two lower dimensional probability distributions: one which acts as a clustering mechanism, and the other which acts as distributional semantic representation $\mathbf{f}_w$.

In this work, we investigate a probabilistic approach to learning $\mathbf{f}_w$. In the spirit of a completely unsupervised scenario, we use an unlabelled dataset to evaluate the methods. We introduce the model architecture in Section 2 in which we describe step-by-step how to progress from a completely unsupervised dataset to calculating a relevance score for new documents, given a corpus of special interest.. Section 3 provides context on the application and dataset. We motivate the experimental setup, evaluation metrics and results in Section 4 and conclude the paper with conclusions and future work in Section 5.

2 Semantic similarity architecture

In this section we break down the tasks involved in establishing a semantic similarity score as shown in the architecture in Figure 1. The white blocks represent input or output artefacts, such as corpora, matrices or scores. The embedded grey blocks represent an algorithm or calculation. The dashed grey blocks indicate the respective subsections explaining the architecture in Section 2.

We assume a preprocessed corpus in this architecture, i.e. all special characters and stop words are removed. The specific preprocessing pertaining to long and short text is discussed in Section 3. In order to test the DSM's generative abilities, we split the corpus into a training and test set.

2.1 Distributional semantic model

The first task in the architecture is to train the DSM on the training corpus. In this paper we consider two DSMs: LDA and GPM for which we provide brief technical overviews below.

Latent Dirichlet Allocation Owing to its long history of success in modelling topics, LDA [1] is arguably the most popular topic model. It is three level hierarchical Bayesian model which models the assumption that the corpus is formed through the following generative process [1]:

1. Assuming the corpus consists of K topics, randomly choose topic distributions $\phi_k \sim Dirichlet(\beta)$ for $k = 1, 2, \ldots, K$.
2. For each of the M documents, $\mathbf{w}_m = \{w_{m1}, w_{m2}, \ldots, w_{mN}\}$, randomly choose a distribution of topics, $\theta_m \sim Dirichlet(\alpha)$.
3. For each of the N words, w_{mn}:
   a) Randomly choose a topic assignment, $z_{mn} \sim Multinomial(\theta_m)$.
   b) Randomly choose a word, $w_{mn} \sim Multinomial(\phi_{z_{mn}})$.

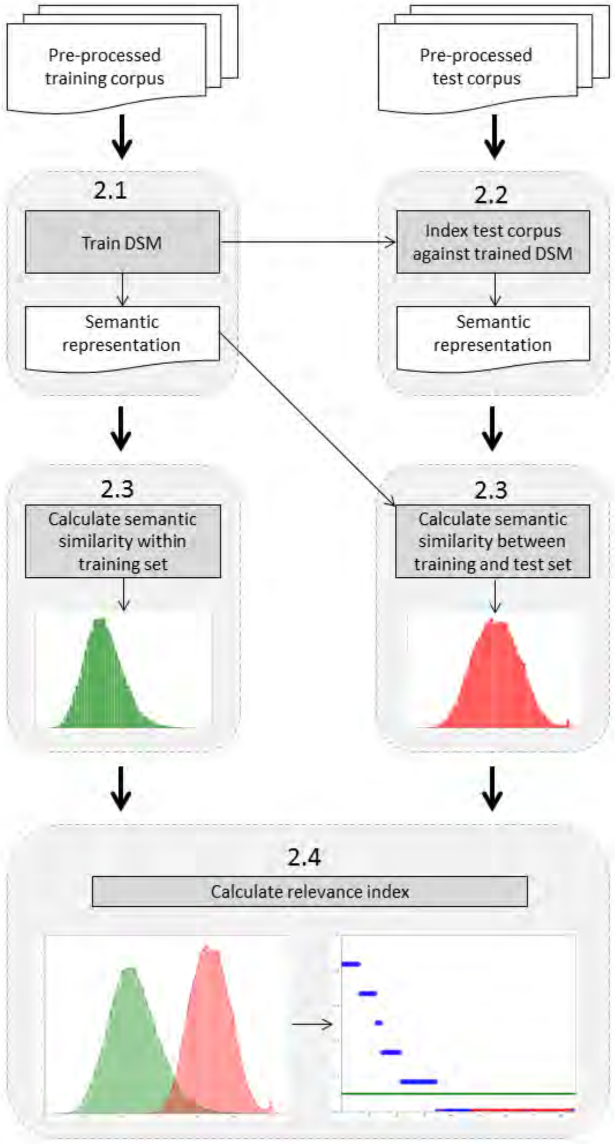

Pre-processed training corpus
Pre-processed test corpus
2.1
Train DSM
Semantic representation
2.2
Index test corpus against trained DSM
Semantic representation
2.3
Calculate semantic similarity within training set
2.3
Calculate semantic similarity between training and test set
2.4
Calculate relevance index

Fig. 1: Semantic similarity architecture

The marginal distribution of a document is thus given by

$$p(\mathbf{w}_m) = \int p(\theta_m \mid \alpha) \left(\prod_{n=1}^{N} \sum_{z_{mn}} p\left(z_{mn} \mid \theta_m\right) p\left(w_{mn} \mid \phi_{z_{mn}}\right) \right) d\theta_m. \tag{1}$$

The posterior distribution of the latent variables is intractable, therefore calling for approximation using either variational inference or Gibbs sampling.

Gamma Poisson Mixture (GPM) The GPM [8] is also a hierachical Bayesian model for modelling topics in short text. Unlike LDA which assumes each documents in a corpus contains multiple topics in different proportions, the GPM assumes that each document contains exactly one topic. Whilst the GPM assumption may seem rigid or unrealistic for long text, such as journal articles and ebooks, it has been shown that this assumption is appropriate for some short texts, such as tweets [7, 16]. The GPM assumes the following probabilistic generative process for a document:

1. Assuming the corpus consists of K topics, a topic k, is first randomly selected according to the mixing weights, $p(z = k)$ where $\mathbf{z} \sim Multinomial(\gamma)$.
2. Each of the M documents, $\mathbf{x}_m$, is then generated based on the the randomly selected topic from the distribution $p(\mathbf{x}_m | z = k)$.

Consequently, the likelihood of a document is given by

$$p\left(\mathbf{x}_m\right) = \sum_{k=1}^{K} p\left(\mathbf{x}_m \mid z = k\right) p(z = k) = \sum_{k=1}^{K} \prod_{v=1}^{V} p\left(x_{mv} \mid \lambda_{kv}\right) p(z = k), \tag{2}$$

where x_{mv} denotes the number ot times word v in the vocabulary occurs in document m for $m = 1, 2, \ldots, M$ and $v = 1, 2, \ldots, V$. Furthermore, it is assumed that $x_{mv} \sim Poisson(\lambda_{kv})$, where the parameter λ_{kv} represents the expected frequency of word v in topic k for $k = 1, 2, \ldots, K$. Each λ_{kv} is then estimated using a collapsed Gibbs sampler and each topic is then described by the words associated with the highest estimates.

2.2 Test set word embeddings

The preprocessing of the training set results in a vocabulary of unique words and in the case of topic models, with a bag-of-words vectorisation of the unstructured documents. The test set may contain words which are not present in the training set vocabulary. Therefore, we need to transform the test set to the same bag-of-words vectorisation than the training set. Now the test corpus is indexed against the trained DSM. In practice, this implies the calculation of a vector $\mathbf{f}_w$ for each test document, based on the trained DSM.

2.3 Semantic Similarity

In the input documents provided to the models, there are words that are similar to each other. Not only in the sense that there are synonyms of some words in the corpus, but that words are related to each other because they pertain to a similar usage. For example, “cat” and “dog” are not synonyms but they are similar in that they can be classified as coming from a common topic i.e. “mammals” or “pets”, etc. Word similarity is very useful in larger semantic tasks such as comparing two word vectors to each other to determine whether or not the words in the vectors are similar to each other. Finding the semantic similarity of two vectors falls in the field of information retrieval (IR). That is, IR is the task of finding document d from the D documents in some collection that best matches some query q [5].

To do this, we will use a probabilistic measure i.e. the Jensen-Shannon distances to evaluate the topic model DSMs. This is because the output of the LDA model yields doc-topic distributions of the corpus.

Jensen-Shannon Distances The first probabilistic analog for vector-space similarity measures that comes to mind is the the Kullback-Leibler (KL) divergence. The KL divergence is not very useful as a metric since it is asymmetric. However, one symmetric version of the KL divergence is the JS divergence defined as,

$$JS(P,Q) = 0.5\boldsymbol{KL}(P||M) + 0.5\boldsymbol{KL}(Q||M)$$

where $M = 0.5(P+Q)$. The JS distance is then calculated by taking the square-root of the JS divergence. The smaller the JSD, the more semantically similar the documents because the documents have similar distributions (in accordance with the distributional hypothesis) [11].

2.4 Relevance Index

The JS distance provide us with a metric to calculate the semantic similarity between documents. To determine how well the models performed in being able to distinguish between semantically different text, we calculated probabilities based on how likely it would be that some new document came from a given reference corpus, given the JS distances between this document and the documents in the reference corpus. We use the same Introduction example to explain our logic:

After digging through thousands of corpora pertaining to ‘Discovery’ you have isolated a corpus of some 3000 documents that relate exactly to the company Discovery Ltd which we shall denote as the reference corpus in this instance. However, you are given new data and you want find out whether or not those new documents are semantically related to the reference corpus. One way to do that is by calculating the mean similarity of each document *within* the reference corpus. It makes sense to assume shorter distances between documents within a reference corpus. Having 3000 documents in the reference corpus, the output of this calculation is a histogram of 3000 values which can be visualised as a histogram such as the one in the lower left block of Figure 1. Secondly, we calculate the mean similarity between the new document and the reference corpus. Now we can ask the question: "How likely is it to obtain this semantic similarity or higher from the reference set?". Using this we can develop a relevance index where high probabilities indicate high relevance and vice versa.

3 Application

In light of the COVID-19 pandemic as of writing, the authors of this paper saw it fit to contribute to the COVID-19 research being conducted at this point. The COVID-19 Open Research Dataset (CORD-19) is a growing corpus of medical papers that was launched in March 2020 by the Allen Institute for AI and their partners. The database initially contained 28000 papers, however it has expanded to well over 140000. The aim of *CORD-19* is to intertwine machine learning with medical research to help policy makers globally deal with the pandemic, by harnessing the invaluable information contained within the dataset. To this end, many online NLP tools have been created to assist governments with domestic policy for containment of the virus. The type of papers contained in the dataset comes from an array of biological science subfields like virology, immunology, genetics, pathology and others [15] A Python package, `cotools`, was developed by this consortium in order to facilitate extraction of the data which is stored in full text JSON format. We consider two datasets for the purpose of evaluating DSMs for short text and long text.

3.1 Long text

CORD-19 contains two datasets - 'comm_use_subset' and 'noncomm_use_subset'. The 'comm_use_subset' corpus contains approximately 20000 full text papers and the 'noncomm_use_subset' corpus has approximately 5000 papers. We used 'comm_use_subset' as the training set for LDA whereas 'noncomm_use_subset' was used for testing and generating the relevance index.

Prior to the application of the models, both corpora underwent standard preprocessing. Uppercase words were reduced to lower case, stop words, numbers and special characters were removed and then the words were stemmed. Since LDA does not perform well on short text, all the documents containing less than 40 words were excluded. This resulted in a training corpus of 19213 documents and a test corpus of 4687 documents of which are partitioned into smaller reference and query sets. The vocabulary size is 361886 unique words.

3.2 Short text

The experiments on the short text were conducted on the abstracts associated with the 'comm_use_subset' and 'noncomm_use_subset' papers. We will refer to these short text corpora as the 'comm_abstracts' corpus and 'noncomm_abstracts' corpus. Since, some papers did not have abstracts, the 'comm_abstracts' and 'noncomm_abstracts' corpora only contained 8750 and 1830 documents, respectively. The corpora underwent the same preprocessing as was done on the long text and the average length of the documents in the 'comm_abstracts' and 'noncomm_abstracts' corpora was 104 and 92 respectively.

In [8], GPM was applied to cleaned text that averaged 8 to 15 words per document. In light of this, prior to the application of the GPM we truncated each document and only kept the first 10 words of each document. As opposed to randomly selecting 10 words form each documents or using more sophisticated feature selection methods, such as tf-idf, this procedure ensures that the selected words originate from the first few sentences. Thus, if the preprocessing was to be reversed, the selected words would be found in coherent sentences. In addition, we also deemed it sensible to assume that most authors are likely to give careful consideration to the first few sentences in their abstracts, thus we selected the first 10 words rather than 10 words from elsewhere in the abstracts. This resulted in a vocabulary containing 9777 unique words.

4 Experiment

The objective of the experiments is to evaluate two probabilistic DSMs on its ability to distinguish between semantic similar and dissimilar text. Because the documents in the dataset are unlabelled, we first had to label it. To determine labels, we train the topic model on the whole corpus and use the resulting dominant topic of each document as the label. We will discuss the use of these document labels later on in this section. The experimental workflow follows the steps in Figure 1 and we briefly reiterate it below.

1. Train a probabilistic DSM on the training set as specified in Section 3. This will yield a semantic representation of the training set.
2. Select a reference set - this is a collection of documents that we know are semantically similar (as our 'Discovery' example).
3. Index the reference and query set.
4. Calculate the semantic similarities within the reference corpus.
5. Calculate the semantic similarities between the reference and query sets.
6. For each document in the query set, calculate the probability of the semantic similarity measure being obtained from the reference set semantic similarities.
7. Because the documents are labelled, we can colour code the documents in the query set: Blue represents documents from the reference set and red represents documents from other sets.

Good performance indicators for the DSMs are high relevance indexes for 'blue' documents and lower relevance indexes for 'red' documents. We go further and put a threshold on the query set relevance indexes in order to calculate hard classification metrics such as accuracy, precision and recall. 'Red' documents above the threshold are counted as false positives and ‘blue’ documents below the threshold are counted as false negatives. We conduct two experiments, experiment 1 compares DSMs on long text and experiment 2 compares DSMs on short text. We discuss the experimental design for each experiment in the next subsections.

4.1 Experiment 1: Long text

The first step in experiment 1 was to extract labels for the documents using LDA which inferred 6 topics. To make some inferences into what these topics consisted of, we reviewed the top 10 words in each topic. An excerpt of the topics is found below:

- **Immunology** [‘cell’, ‘use’, ‘antibodies’, ‘infect’, ‘figure’, ‘virus’, ‘protein’, ‘vaccine’, ‘incubation’, ‘assay’]
- **Genetics** [‘sequence’, ‘use’, ‘sample’, ‘gene’, ‘detect’, ‘genome’, ‘analysis’, ‘virus’, ‘differ’, ‘result’]

We chose to use topic 1 and topic 4 from the model for testing which we determined to be about ‘Immunology’ and ‘Genetics’ respectively. It must be noted that, in topic modelling, the labelling of topics must be conducted manually, hence it is subjective and a different user may arrive at different labels [4]. We then indexed the test set against the trained LDA model and we split the training and test sets according to each document’s dominant topics. Thereafter, we plotted histograms of semantic comparisons between these topics respectively, using the adjusted Jensen-Shannon distances AJSD (where AJSD = 1-JSD) to compare the topic-distributions for LDA. High values of the AJSDs indicate high similarity whereas low values indicate low similarity. These histograms are shown in Figure 2.

From this, we see the results of our comparisons for LDA. The purple histogram indicates the distances between documents within the reference set ‘Genetics’. This distribution peaks around 0.65 and is relatively distinct from the yellow histogram which indicates the distances between documents from the reference set ‘Genetics’ and the alternative set ‘Immunology’. We find to this end, that this yellow histogram peaks around 0.3 which is indicative of low semantic similarity. Therefore, in this regard, the LDA algorithm has proven itself capable in its ability to distinguishing semantically different and similar text.

Given the average distance between an new test document and the reference set, we can calculate the probability of the semantic similarity measure being obtained from the reference set semantic similarities, which we call the relevance index. Because we have labelled documents, we can extend our evaluation to classification metrics and determine the accuracy, precision and recall of each model. We do this by setting a decision threshold for classification. Figure 3 shows the relevance indexed for a held-out test set containing relevant (blue)

[4] It should be noted that the authors of this paper are not medical experts. However, a medical doctor was consulted to assist with the identification of the topics.

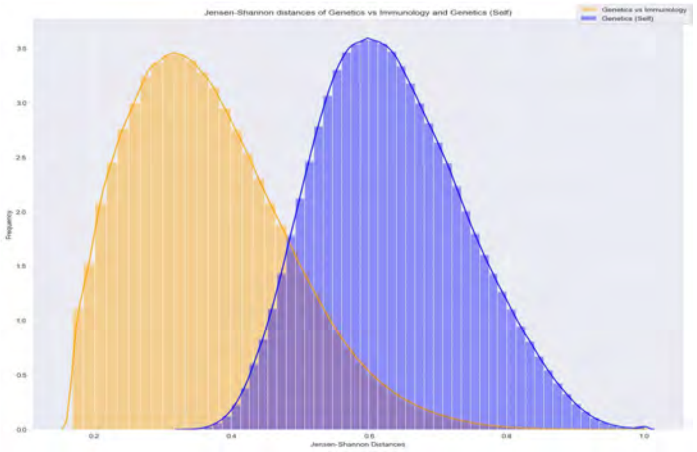


Fig. 2: Histograms of adjusted JS distances within the reference corpus (purple) and between reference and alternative corpus (yellow). The DSM in this case is LDA.

Metric	CORD-19 Long Text	CORD-19 Abstracts	
	LDA	GPM	LDA
Precision	1.0	1.0	1.0
Recall	0.8991	0.8742	0.8118
Accuracy	0.9481	0.9309	0.9169

Table 1: Evaluation of models on different corpora. In all cases a decision threshold of 0.1 was set.

and irrelevant (red) documents. A decision threshold of 0.1 was set in this case, indicated by the green line. Table 1 shows the precision, recall and accuracy statistics for each model.

From Table 1 under CORD-19 Long Text and Figure 3, it is very clear that the LDA model performs very well in determining semantic similarity of text. The LDA model correctly classified all the irrelevant documents below the threshold . This is backed by the precision scores of LDA which is 100%. This means that LDA correctly predicted true positives 100% of the time. The accuracy of LDA is 0.9481 meaning that LDA correctly predicts relevant documents in the right classes almost 95% of the time.

4.2 Experiment 2: Short text

We extend Experiment 1 to the short text dataset. We have established a good baseline for LDA in Experiment 1, and compare it with GPM, which is a tai-

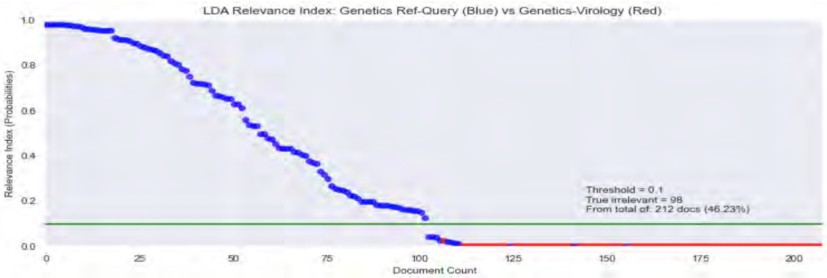


Fig. 3: Relevance indexes of test documents: LDA applied to *CORD-19*

lored topic model for short text. In order to provide a fair comparison, we label documents using the topic model under investigation, i.e. if we test GPM, we use GPM to generate the labels. GPM generated 10 topics of which an excerpt of 2 topics is found below:

- **Virology** [‘use’, ‘sequence’, ‘study’, ‘method’, ‘gene’, ‘detect’, ‘protein’, ‘develop’, ‘genome’, ‘virus’]
- **Pulmonology** [‘respiratory’, ‘coronavirus’, ‘syndrome’, ‘middle’, ‘east’, ‘severe’, ‘merscov’, ‘cause’, ‘acute’, ‘human’]

LDA generated 6 topics on this short text of which 2 topics were also indicated to be related about ‘Virology’ and ‘Pulmonology’:

- **Virology** [‘virus’, ‘viral’, ‘cell’, ‘protein’, ‘response’, ‘host’, ‘rna’, ‘immune’, ‘role’, ‘include’]
- **Pulmonology** [‘respiratory’, ‘syndrome’, ‘coronavirus’, ‘severe’, ‘acute’, ‘pneumonia’, ‘associate’, ‘middle’, ‘east’, ‘detect’]

Experiments to compare semantic similarities between these topics were conducted and the results are summarised in Figure 4. The results for GPM are shown in the top graph whilst the bottom graph shows the LDA results. As the GPM assigns documents to topics in a manner akin to hard clustering, the distances tend to be extremely small or extremely large. For GPM, the average of the adjusted Jensen-Shannon distances between the reference and query set of ‘Virology’ (green) is 0.932 (median 0.995) thus indicating a high similarity between documents. In contrast, the comparison between the reference set of ‘Virology’ and the query set of ‘Pulmonology’ (red) indicates low similarity between the sets. The average of the adjusted Jensen-Shannon distances 0.178 (median 0.167).

Unlike GPM whihc produces extreme values, LDA assigns different topic proportions per topic for each document, thus there is greater variation in the values of the distances. For LDA, the average of the adjusted Jensen-Shannon distances between the reference and query set of ‘Virology’ (green) is 0.820 (median 0.819) and the average disances between ‘Virology’ and ‘Pulmonology’ is 0.655 (median

0.649). It is interesting to note that the distance between the means of the two histograms for GPM is much larger (0.754) than that of the histograms from LDA (0.165). We can also observe from the LDA graph that there is some overlap between the histograms. If we focus on the histogram between 'Virology' and 'Pulmonology', this overlap indicates that there are some documents within these differing sets that have similar semantic representations.

In a similar manner to what was done in Section 4.1, we compiled a similarity index to determine the accuracy, precision and recall of each model. The results are summarised in Figure 5 in Table 1. The reference set was made up of the 'Virology' set from the 'comm_abstracts' corpus. We took the 'Virology' set from the 'noncomm_abstracts' as our semantically similary query set and then the 'Pulmonology' set from the 'noncomm_abstracts' corpus as our semantically dissimilary query set. From Figure 5 and Table 1 under CORD-19 Abstracts, it can be seen that GPM outperforms LDA with respect to accuracy, precision and recall.

5 Conclusions

In this paper we investigated the performance of two probabilistic DSMs on a medium sized corpus. We tested the model's ability to succesfully assign a high relevance index for a test document which is semantically similar to a reference corpus. We showed that probabilistic DSMs are able to sucessfully assign appropriate relevance indexes to unseen documents, based on the relevant corpus of interest. We expanded the investigation into long and short text and furthermore showed that a probabilistic DSM designed for short text - the GPM model - performs best on short text.

Apart from the experimental results, we also introduced an experimental workflow for evaluating DSMs. An important contribution is the definition of relevance index, which is a normalized performance metric to compare different similarity measures.

The experiments were done on completely unstructured and unlabelled datasets. In order to accommodate this, we used topic models to categorise and label the existing corpora.

Future work include an expansion of experiments - including a variety of corpus genres as well as other word embeddings such as tf-idf and GloVe.

References

1. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. The Journal of Machine Learning Research **3**, 993–1022 (Mar 2003)
2. Brody, S., Elhadad, N.: An unsupervised aspect-sentiment model for online reviews. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. pp. 804–812. Association for Computational Linguistics (2010)

3. Fefferman, C., Mitter, S., Narayanan, H.: Testing the manifold hypothesis. Joural of the American Mathematical Society **29**(4), 983–1049 (2016). https://doi.org/10.1090/jams/852, `http://dx.doi.org/10.1090/jams/852`
4. Hofmann, T.: Latent semantic models for collaborative filtering. ACM Transactions on Information Systems (TOIS) **22**(1), 89–115 (2004)
5. Jurafsky, D., Martin, J.H.: Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Third edn. (2019). https://doi.org/10.1515/zfsw.2002.21.1.134
6. Levy, O., Goldberg, Y., Dagan, I.: Improving distributional similarity with lessons learned from word embeddings. Transactions of the Association for Computational Linguistics **3**, 211–225 (2015)
7. Mazarura, J., De Waal, A.: A comparison of the performance of latent dirichlet allocation and the dirichlet multinomial mixture model on short text. In: 2016 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech). pp. 1–6. IEEE (2016)
8. Mazarura, J., de Waal, A., de Villiers, P.: A gamma-poisson mixture topic model for short text. Mathematical Problems in Engineering **2020** (2020)
9. McCallum, A.: Multi-label text classification with a mixture model trained by em. In: AAAI workshop on Text Learning. pp. 1–7 (1999)
10. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. 1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings pp. 1–12 (2013)
11. Murphy, K.P.: Machine Learning: A Probabilistic Perspective. MIT Press, Cambridge, Massachusetts, first edn. (2012). https://doi.org/10.1007/978-94-011-3532-0_2
12. Ó Séaghdha, D., Korhonen, A.: Probabilistic distributional semantics with latent variable models. Computational Linguistics **40**(3), 587–631 (2014)
13. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp. 1532–1543 (2014)
14. Wang, C., Blei, D.M.: Collaborative topic modeling for recommending scientific articles. In: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 448–456. ACM (2011)
15. Wang, L.L., Lo, K., Chandrasekhar, Y., Reas, R., Yang, J., Burdick, D., Eide, D., Funk, K., Katsis, Y., Kinney, R., Li, Y., Liu, Z., Merrill, W., Mooney, P., Murdick, D., Rishi, D., Sheehan, J., Shen, Z., Stilson, B., Wade, A.D., Wang, K., Xin, N., Wang, R., Wilhelm, C., Xie, B., Raymond, D., Weld, D.S., Etzioni, O., Kohlmeier, S.: CORD-19: The COVID-19 Open Research Dataset. In: Proceedings of the Workshop on NLP for COVID-19 at ACL 2020. Association for Computational Linguistics (2020), `https://arxiv.org/abs/2004.10706`
16. Zhao, W.X., Jiang, J., Weng, J., He, J., Lim, E.P., Yan, H., Li, X.: Comparing twitter and traditional media using topic models. In: European conference on information retrieval. pp. 338–349. Springer (2011)

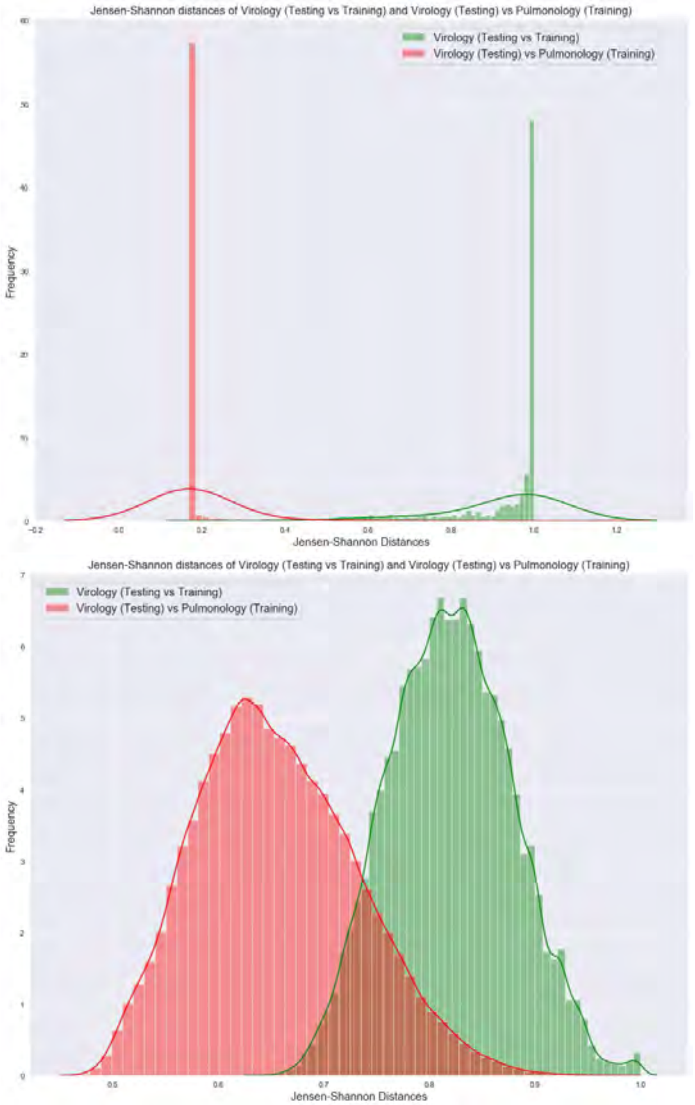


Fig. 4: Semantic comparisons between topics for GPM (top) and LDA (bottom) on short text

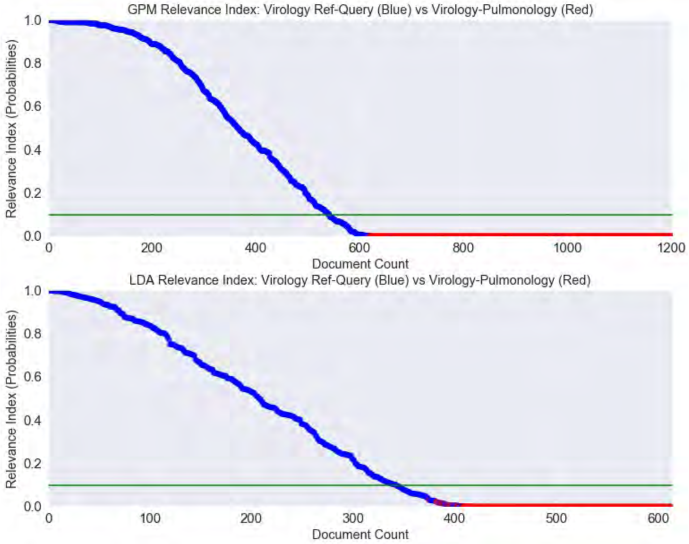


Fig. 5: Relevance Index: GPM and LDA applied to *CORD-19* abstracts

Machine Learning for Improved Boiler Control in the Power Generation Industry

Tshidiso C. Mazibuko[1][0000−0003−1423−9090] and Katherine M. Malan[1][0000−0002−6070−2632]

Department of Decision Sciences, University of South Africa, Pretoria, South Africa
malankm@unisa.ac.za

Abstract. In the South African context, steam boiler failures in power generation plants come at a huge cost to the gross domestic product (GDP) of the country. Load shedding, largely a result of steam boiler failures causing plant shutdown, resulted in a loss of a minimum of R59 billion in 2019. Even with the installation of control systems, steam boilers are prone to failures. In this paper, we provide a survey of examples of the application of machine learning models in predicting steam boiler failures and optimising the performance of steam boilers with the aim of avoiding common failure mechanisms such as overheating. These examples of the application of machine learning models in steam boilers could be applicable to South Africa's power generation plants that are prone to failures which cause power outages. We also replicate a study that uses machine learning to predict temperatures at different zones inside a boiler. These predictions are used as the basis for controlling the boiler to maintain a uniform temperature distribution (and as a result prevent the high occurrence of tube leak failures caused by non uniform temperature distribution). We show how feature selection can significantly improve the performance of the machine learning model, while simultaneously simplifying the model. Simpler models are not only easier for decision-makers to understand but are also less computationally intensive to re-train and implement in real-time systems.

Keywords: Machine learning · Power generation · Steam boilers · Feature selection.

1 Introduction

Most power plants in the world use steam as a medium that is used to turn a turbine for electricity generation. One of the reasons for using steam is that water is one of the cheapest and the most abundant source we can use. Steam for power plants (and other chemical plants that use steam as a utility for heating) is generated by boilers. Steam boilers are prone to failure, which can result in power outages.

The reported availability of power plants in South Africa, accounting for planned and unplanned shutdowns, was 67% for 2019 [27]. The economic cost to the gross domestic production (GDP) as a result of this unavailability (load shedding in South African context) was estimated by the Centre for Scientific and Industrial Research (CSIR) to be between R59 and R120 billion in 2019 [27]. Unavailability of steam boilers is predominantly due to unplanned shutdowns rather than planned shutdowns. The failure mechanisms that lead to unplanned shutdowns vary. According to a study conducted in South Africa by McIntyre [20], on the failure mechanism of steam boilers fuelled by biomass and other sources such as coal and heavy fuel oil, five mechanisms were identified as the causes of unplanned shutdowns:

1. Corrosion: deposit of solid material inside tubes,
2. Erosion: deposition of suspended ash particulates on the outside tubes causing thinning of the tube material, leading to cracks or tube leak when the pipe is exposed to high temperatures,
3. Fatigue (thermal, mechanical and corrosion): largely caused by cyclic operations (cyclic load requirements) and a combination of corrosive environment, mechanical stress on tubes,
4. Stress corrosion cracking: Corrosive environment and sustained tensile stress, and
5. Overheating: caused by pipes exposed to very high temperatures or non uniform temperature distribution and causing tube rupture.

Overheating accounted for 44% and 57% of the failure modes in water tube boilers for non biomass power plants (e.g coal fired) and biomass power plants respectively, followed by erosion/corrosion accounting for 23% of the failure modes [20]. Uneven temperature distribution inside the boiler is one the causes of unstable operation of the boiler, in addition it is one of the main causes of tube leaks [4]. Maintaining uniform temperature distribution is a difficult task even with the application of advanced process control.

This paper describes four recent studies that used data-driven approaches with machine learning to improve the operation of steam boiler systems. The purpose is to highlight some of the ways in which machine learning could be used to support the industry. In addition, one of the studies involving the prediction of temperatures within zones of the boiler is replicated as a case study. It is shown that through feature selection, the model can be simplified, while simultaneously improving the predictive power of the model.

2 Steam Boiler System Overview

A steam boiler in a thermal power plant is one of the most important unit operations. Essentially this unit operation is responsible for converting water to steam at supercritical conditions and the steam is used to turn a turbine that eventually produces electricity. The boiler unit operates by combusting fuel (e.g coal) with air in a combustion chamber to produce hot flue gas. The energy in the hot flue gas is transferred to the water wall tubes by radiation and convection, wherein water is converted to saturated steam. The saturated steam is directed to a steam/water drum, and the steam from the steam/water drum is directed to a superheater where the flue gas temperature is further reduced by exchanging energy with saturated steam to produce superheated steam. The superheated steam is sent to the turbine for electricity generation. The flue gas temperature is further reduced by exchanging energy with the feed water through the economiser. The hot water is directed to the steam/water drum before it flows into the water wall tubes. The flue gas from the economiser passes through the fabric filters (to remove solid particles) and is finally sent to the stack.

Steam boilers are fitted with instrumentation linked to a distributed control system to maintain safety and achieve optimum performance. Four important variables are normally controlled for boilers:

1. Water-steam drum level: This is important for preventing carrying over of water into the steam lines increasing the risk of downstream equipment damage and preventing exposing water wall tubes from excessive heat.
2. Fuel flow: This is important for combustion control and the quality of the steam produced.
3. Air flow: This controls combustion and furnace pressure.
4. Flue gas oxygen content: This controls emissions of nitrogen oxide gases and thermal efficiency of the boiler.

3 Machine Learning for Improving Steam Boiler Systems

This section describes four recent studies from the literature in which machine learning has been used to improve the performance of steam boiler systems.

3.1 Biomass Steam Boiler Root Cause Analysis

Laubscher *et al.* [16] compared the performance of different machine algorithms in finding the root cause of excessive final steam temperature for a biomass steam boiler. The case study is based on a sugar-producing plant in South Africa that uses biomass by-product from its operations (sugar cane bagasse and furfural residue), as fuel in a steam boiler to generate power for internal use. The operation of this boiler was experiencing final steam temperatures up to 35% above the design limit of 100 °C. The high final steam temperature can cause metallurgical failure of the superheater, leading to the shutdown of the boiler and can also lead to failure of downstream equipment such as the turbine. The end goal of the study was to develop a predictive model to explain causes of high final steam temperature.

In the building of the model, 28 process variables were considered as independent variables and the target variable was the final steam temperature. The independent variables were compressed to 15 variables through the use of data compression technique and further compressed to 8 variables by averaging the 7 fuel feeder speed variables into one fuel feeder speed variable. The two fuel chute densities were also averaged. Three models, artificial neural networks (ANN), support vector regression (SVR) and random forest (RF) were trained and tested. In determining the variables that have significant impact on steam temperature, sensitivity analyses using the machine learning models were performed. The sensitivity analyses revealed the following:

- High steam temperatures occur at low fuel moisture content. Low fuel moisture content means high fuel energy content and high flue gas temperature (and eventually led to an increased heat transfer over the steam superheater).
- High steam temperatures were more pronounced at high fuel densities (i.e. low fuel particle size). Low particle size causes the burner flame to be positioned high up the furnace and close to the superheater. This results in high heat transfer rate over the superheater above the design rate.
- An increase in the forced draught fan damper position (which controls flow of air) caused an increase in the final steam temperature. This was more pronounced at high feed chute densities (i.e. low fuel particle size).

All the three models showed the same behavioural pattern when sensitivity analysis was performed. The next phase of the study was to use the machine learning model as part of the control logic that would lower the fan damper position at high chute densities and increase the fan damper position at low chute densities.

3.2 Steam Boiler Pipe Leak Detection System

Steam boilers in thermal power plants are prone to pipe leaks mainly due to fatigue and thermal stresses. Pipe leak failures are incipient in nature and gradually become enlarged with time. They lead to loss of productivity, secondary failure of other equipment and unplanned shutdowns. Hence it is important to have techniques for predicting pipe leak failure in advance to avoid unnecessary plant shutdowns and most importantly avoid catastrophic safety incidents. Commonly used industrial techniques for predicting pipe leaks are [23]:

1. Sensor-based acoustic methods which measure acoustic waves from the leaking steam,
2. Mass balance of the water/steam system,
3. Measuring of humidity of the flue gas, and
4. Methods based on monitoring of other process variables supported by process models.

Acoustic techniques require the use of specialised and expensive sensors and generally do not detect leaks less than 10 000 kg/hr. Acoustic methods will not detect early stages (incipient) of a pipe leak. The mass balance technique is insensitive to small leaks. Monitoring of humidity of the flue gas is often negatively affected by blowing of soot inside the boiler. The use of a process model requires very high levels of accuracy in order to detect pipe leaks for boilers and is difficult to attain.

For these reasons Swiercz [23] used a data driven approach in the form of principal component analysis (PCA) to detect failures due to pipe leaks. In the study, 25 cases of shutdowns caused by pipe leaks were analysed for a coal-fired power plant in Poland. Twelve variables, with inputs from the operators and technical personnel, were selected to prepare a PCA model for a 'healthy' system. These variables were shown to be very sensitive to tube leaks. The principal components were narrowed down to three components. Two statistics values, T^2 and squared prediction error (Q) were used to determine data points outside the confidence ellipsoid and those data points would be regarded as data points that point to a tube leak. The data-driven model achieved the following:

1. 60% of the 25 cases of shutdowns caused by pipe leaks could be detected 3-5 days before the emergency shutdown.
2. Only 6% of the cases could be detected in less than day before emergency shutdown.
3. Data points that were 8-10 days before shutdown were proven to be within the confidence ellipsoid.

The benefits of early detection are the avoidance of extra repairs of equipment, short period of production outage and planning for planned shutdown. All these benefits have a direct link to the improvement of economics of production.

3.3 Steam Boiler Combustion Efficiency Optimisation and Virtual Testing

Kusiak and Song [15] used a data-driven approach in the form of a clustering method to search for control settings (controlled parameters such as coal feeder speed, flow of air into the boiler, etc.) for improving boiler efficiency. In their study, an industrial power plant historic data set of controllable and non controllable parameters (such as temperature of inlet boiler feed water) was collected over a period of time with their corresponding efficiency levels. The K-means algorithm was used to create clusters of data points (with a discrete range of efficiency corresponding to each cluster) and these clusters were stored in a knowledge base. For a future data point, the knowledge base was searched for controllable parameters of a cluster with efficiency greater than the current efficiency of the data point. The parameters of the found cluster would then be proposed as the control settings to improve the efficiency of the boiler.

The proposed control settings from the clustering algorithm were tested on a simulation of the boiler (virtual boiler) designed using a machine learning model. The inputs to the machine learning model were historic data points of controllable and non controllable parameters and the output was the boiler efficiency level. The proposed control settings from the K-means algorithm were used as the input to the ML model to test whether there was an improvement in the efficiency of the boiler. Several data points proved that this approach indeed yielded improved efficiency levels. Two machine learning models for the virtual boiler were compared and the decision tree model showed better prediction results than the ANN model.

3.4 Optimising Boiler Control in Real Time

Maintaining uniform temperature distribution at different zones inside a boiler and maintaining consistent oxygen content of the flue gas improve the stability and energy efficiency of the boiler [4]. In addition, the non uniform temperature distribution inside the boiler is one of the main reasons boilers experience pipe leaks (due to thermal stresses caused by uneven temperature distribution) and eventually causes boiler instability [4]. Maintaining balanced, low oxygen levels of the flue gas also implies high efficiency levels. Modelling the temperature distribution inside the boiler is a difficult task and the traditional approach has been the use of computational fluid dynamics (CFD). The combustion process is dynamic in nature and CFD analysis models steady state processes and also cannot be used for real time process control as it is computationally expensive.

Ding *et al.* [4] propose an approach of integrating machine learning methods and optimisation techniques in an attempt to achieve uniform temperature distribution inside the boiler and uniform oxygen levels of the flue. The problem was approached in the following manner:

1. Historical data of controlled variables and uncontrolled variables was collected using the distributed control system. Included in the data was collection of temperature at six different zones and oxygen content at both sides of the flue.
2. A machine learning model was built for predicting temperatures at six different zones and oxygen content at two sides of the flue using historical data.
3. An optimisation model was built with objective of maintaining optimal temperature distribution within the boiler, based on the predicted temperatures and oxygen levels from the machine learning model. The output of the model was the optimised values of controlled variables to be used as inputs to the boiler control scheme.

Four machine learning methods were tested for the prediction task, namely SVR with linear kernel, SVR with non-linear kernel, classic three layer feed forward ANN, and a recurrent neural network (RNN). The linear SVR yielded good results for temperature prediction with lower mean squared error (MSE) compared to the other three methods. For oxygen prediction, the ANN yielded better results than linear SVR. However, the linear SVR model was chosen over ANN due to efficiency in real time processing. Five different optimisation algorithms were tested: interior point method (IP), genetic algorithm (GA), particle swam optimisation (PSO), sequential quadratic programming (SQP) and differential evolution (DE). IP produced better results in the reported run time, lower mean for the objective function (since the optimisation model is executed at every time step) and lower standard deviation for the objective function.

The overall result of the use of the optimisation model using IP algorithm and the SVR method yielded a reduction of temperature standard deviation by 42%, and a reduction of the difference of oxygen content from two sides of the flue by 61% when compared to original data without the optimisation model. The average temperature increased by 32% and the average oxygen content decreased by 38.6%.

3.5 Summary

The studies described above show that machine learning models have been applied successfully in fault detection of steam boiler failure modes such as tube leaks, root cause analysis and optimisation of the performance of the steam boiler with the aim of improving boiler efficiency. Many of these approaches could be applicable to South Africa's power generation plants that are prone to failures which cause power outages.

4 Case Study: Boiler Temperature Prediction

Section 3.4 described an approach to optimising boiler control in real time proposed by Ding *et al.* [4]. In this section, the temperature prediction component of the study is replicated and the model is improved through feature selection.

4.1 Dataset

The publicly available dataset[1] contains data collected from a real boiler over a period of more than two months. It has 11301 observations and 80/20 train test split is used. The features are described in Table 1.

Table 1. Description of features in the boiler dataset

Feature	Description
Column 1	Generation load
Column 2	Hot Wind
Column 3	Cold Wind
Column 4	Left NOx gases content
Column 5	Right NOx gases content
Column 6-17	Coal feed rate
Column 18-33	Throttle opening of valves (%)
Column 34	Variable 34

The left and right NOx gases content are the content of the nitrous oxide gases (NOx) on the two opposite sides of the flue system. Variable 34 is a variable whose description is unknown. The dependent variables are the temperatures of the six zones in the boiler.

4.2 Methodology

The approach used in the model development was as follows:

[1] https://github.com/yding5/BoilerOptimization

1. Determine optimal parameters for the machine learning algorithms using hyper-parameter optimisation techniques based on the training dataset.
2. Use the optimal parameters of the machine learning algorithm to replicate temperature prediction results of Ding *et al.* [4].
3. Using the optimal parameters, apply feature selection on the training dataset to determine important features for the machine learning algorithm.
4. Use the test dataset, optimal parameters and the selected features to determine the mean squared error (MSE) and compare it to the MSE computed in step 2.
5. Repeat steps 1 to 4 for each temperature zone.

These steps are described in more detail in the following subsections.

4.3 Parameter Optimisation and Case Study Replication

Ding *et al.* [4] experimented with different machine learning algorithms and approaches to organising the input data for predicting the temperature of the six zones. Our study replicated the radial basis function SVR model approach based on steady state input data with predictor variables only. The parameters for the model included the box constraint (C), epsilon and the kernel scale (gamma). Bayesian optimisation was used to find the optimal parameters for the SVR model for each temperature zone to be predicted and the resulting values are given in Table 2 with the associated MSE values produced by the model.

Table 2. Parameter optimisation results

	Zone 1	Zone 2	Zone3	Zone4	Zone5	Zone6
Box constraint	417	296.84	689.72	99.86	952.40	25.78
Kernel Scale	6.46	6.44	6.81	6.02	7.96	37.16
Epsilon	4.64	0.33	9.65	0.16	1.95	21.36
MSE	1333	1381	1629	1450	1677	997

With the optimised SVM parameters as given in Table 2, the model produced an average MSE over the six temperature zones of 1411 (corresponding mean absolute percentage error MAPE of 2.50%). The corresponding MSE reported in the Ding *et al.* [4] study was 1860 with a MAPE of 2.88%. The difference can be attributed to the fact that hyper-parameter optimisation uses 10-fold cross validation with randomised training and validation sets, and random initialisation in these two cases may be different.

4.4 Feature selection

In most cases, using data for machine learning with many predictor variables (features) is computationally expensive and consumes a lot of memory [17]. This makes the use of machine learning models impractical because of their complexity. They are also not easy to understand. In such cases there are methods that can be used to reduce the number of features so that the machine learning algorithm will only use features that are relevant. One of these methods is the sequential forward selection (SFS) and the aim is to find optimal set of features for the machine learning model [17], while also improving the performance of the machine learning model in terms of speed and accuracy. The main steps of the SFS method are as follows [19]:

1. The SFS algorithm starts with an empty set of features and searches for a feature (out of all features) that yields the lowest value for a criterion function. In the case of regression models, the mean squared error is used as the criterion function. The MSE is calculated using the predicted results of the machine learning model (after being trained) and the actual data.
2. Add/search for another feature to the data that will further reduce the MSE value.
3. Continuously add features such that the MSE of the combined features is reduced until there are no more features that will decrease the MSE value. At this point the algorithm stops.

Other feature selection methods include sequential backward selection, filter methods and genetic algorithms. Sequential backward selection works in the same way as SFS except that is starts with the full set of features and sequentially removes features that do not improve the MSE. Filter methods use the characteristics of features and their relevance to the response variable as a selection methods (e.g. variance, F-tests, correlations amongst features, correlations between features and a response variable, etc). Filter methods are not correlated to a machine learning model [18]. Evolutionary algorithms such as genetic algorithm and particle swam optimisation are also other feature selection algorithms. These algorithms are computationally expensive.

Feature selection using SFS was carried to find the optimal features for the SVR model. The selected features of each zone are presented in Table 3. It can be seen that between 13 (for zone 4) and 21 (for zone 1) features out of the 34 total features were selected by the SFS approach.

One interesting observation in the features selected is that SFS algorithm consistently selects column 1 (generation load) of the dataset in the prediction of the temperatures in the six zones. This is consistent with the energy balance equation that relates the steam production to the temperature inside the boiler. Hot wind and cold wind features (column 2 and 3) are linearly correlated to each other, hence the SFS algorithm rarely selects these features simultaneously (with the exception of zone 1 temperature prediction). Similarly, the left and right NOx gases content (column 4 and 5) are linearly correlated. Hence the SFS algorithm does not select both features simultaneously (with the exception of zone 6 and zone 1). The SFS algorithm consistently does not select column 34 feature (the unknown variable).

The SFS algorithm consistently selects throttle openings of valve 19,27,31,32 and 33. This may indicate the importance of these controlled variables on tem-

Table 3. Features selected by the sequential forward selection method

Column	Zone 1	Zone2	Zone 3	Zone 4	Zone 5	Zone 6
1	1	1	1	1	1	1
2	1	-	-	-	1	-
3	1	1	-	1	-	1
4	1	1	1	1	1	1
5	1	-	-	-	-	1
6	-	-	-	-	-	-
7	-	-	-	-	-	-
8	-	1	1	1	-	1
9	1	-	-	-	-	-
10	-	-	-	-	-	-
11	1	-	-	-	-	-
12	-	-	-	-	-	-
13	-	-	-	-	-	-
14	-	-	-	-	-	-
15	-	-	-	-	-	-
16	1	-	-	1	-	1
17	1	1	-	-	-	-
18	1	-	1	1	1	1
19	1	1	1	1	1	1
20	1	-	1	1	1	-
21	-	1	1	-	1	1
22	1	-	1	-	1	1
23	1	1	1	-	1	1
24	-	1	-	-	1	-
25	-	-	-	1	1	1
26	1	1	1	-	1	-
27	1	1	1	-	1	1
28	1	-	1	1	1	1
29	1	1	-	-	1	1
30	-	1	1	1	1	1
31	1	1	1	1	-	1
32	1	1	1	-	1	1
33	1	-	1	1	1	1
34	-	-	-	-	-	-
Total	21	16	16	13	18	19

perature control. The temperature zones are also dependent on the coal feed rate as there is also a direct energy balance relationship between the temperature in the boiler and the coal feed rate.

4.5 Comparison of the performance before and after feature selection

The next step after feature selection was the computation of the MSE and MAPE of each temperature zone using the selected features and the results are presented in Table 4.

The average MSE and the MAPE values of this case before and after the implementation of feature selection using the SFS algorithm are compared in Table 5.

Table 4. Feature selection results

Zone	MSE	MAPE(%)
1	303	1.16
2	339	1.24
3	303	1.18
4	361	1.28
5	281	1.18
6	366	1.23
Avg	326	1.21

Table 5. Comparison of the MSE and MAPE values

	MSE	MAPE(%)
Before SFS	1411	2.50
After SFS	326	1.21

The application of feature selection improved the accuracy score (MSE) of the SVR RBF model from an average of 1411 to an average score of 326. This shows that feature selection can improve the robustness and accuracy of a machine learning model.

The actual and predicted temperature profiles of selected zones after the implementation of feature selection are shown in Figures 1 to 3. The data in Figure 1 is scattered closely along the 45° line. This shows the machine learning model is approximating the actual temperature of zone 1 satisfactorily well. Similarly for zone 3 and zone 5, the machine learning model approximates the actual temperature well since the data points in Figures 2 and 3 are scattered along the 45° line.

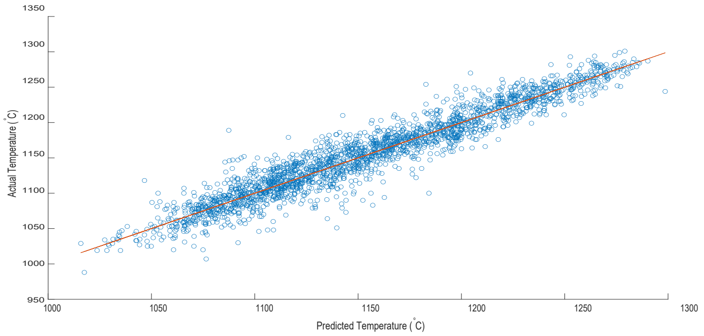


Fig. 1. Comparison of actual temperature and predicted temperature for zone 1

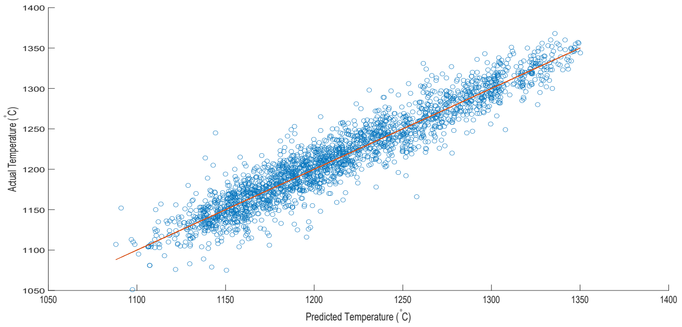


Fig. 2. Comparison of actual temperature and predicted temperature for zone 3

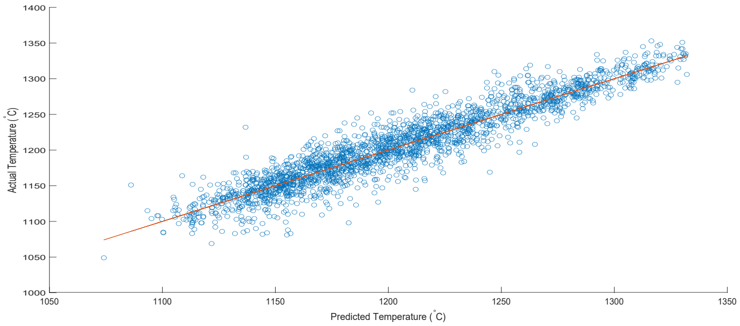


Fig. 3. Comparison of actual temperature and predicted temperature for zone 5

5 Conclusion

This paper provided a review of four recent contributions from machine learning to improving the automated control of steam boilers for power generation. One of the studies involving the prediction of temperature at different zones in the boiler was replicated and improved through feature selection.

There are benefits in using feature selection for a machine learning model. The benefits are:

1. An improved predictive power of the machine learning model which was achieved without the manipulation of features normally employed to improve the performance of the machine learning model.
2. A less complex machine learning model with reduced number of features that can be used in real time optimisation.
3. A machine learning model that will be computationally less expensive to train.

Acknowledgement

This work is based on research supported by the National Research Foundation of South Africa (Grant Number: 120837).

References

1. Aln, F.B.I., AL-Kayiem, H.H.: Artificial intelligent system for steam boiler diagnosis based on superheater monitoring. Journal of Applied Sciences **11**(9), 1566–1572 (Sep 2011). https://doi.org/10.3923/jas.2011.1566.1572, https://doi.org/10.3923/jas.2011.1566.1572
2. Banares-Alcantara, R.: Knowledge-based expert systems: an emerging technology for cad in chemical engineering. Tech. rep., Carnegie Institute of Technology (1984)
3. Beck, D.A.C., Carothers, J.M., Subramanian, V.R., Pfaendtner, J.: Data science: Accelerating innovation and discovery in chemical engineering. AIChE Journal **62**(5), 1402–1416 (Feb 2016). https://doi.org/10.1002/aic.15192, https://doi.org/10.1002/aic.15192
4. Ding, Y., Liu, J., Xiong, J., Jiang, M., Shi, Y.: Optimizing boiler control in real-time with machine learning for sustainability. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management. ACM (Oct 2018). https://doi.org/10.1145/3269206.3272024, https://doi.org/10.1145/3269206.3272024
5. Dunn, K.: Process Improvement using data (2020 (accessed October 10, 2020)), https://learnche.org/pid/
6. E.Hammad, M., Kasban, H., Elaraby, S.M., Dessouky, M.I., Zahran, O., El-Samie, F.E.A.: Distillation column malfunctions identification using higher order statistics. International Journal of Computer Applications **108**(1), 7–13 (Dec 2014). https://doi.org/10.5120/18873-0129, https://doi.org/10.5120/18873-0129
7. Engelbrecht, A.P.: Computational Intelligence. John Wiley & Sons, Ltd (Oct 2007). https://doi.org/10.1002/9780470512517, https://doi.org/10.1002/9780470512517
8. Ge, Z., Song, Z., Ding, S.X., Huang, B.: Data mining and analytics in the process industry: The role of machine learning. IEEE Access **5**, 20590–20616 (2017). https://doi.org/10.1109/access.2017.2756872, https://doi.org/10.1109/access.2017.2756872
9. Gupta', A.E.: Introduction to deep learning part 1 (June 2018)
10. Heo, S., Lee, J.H.: Statistical process monitoring of the tennessee eastman process using parallel autoassociative neural networks and a large dataset. Processes **7**(7), 411 (Jul 2019). https://doi.org/10.3390/pr7070411, https://doi.org/10.3390/pr7070411
11. James, G., Witten, D., Hastie, T., Tibshirani, R.: An Introduction to Statistical Learning. Springer New York (2013). https://doi.org/10.1007/978-1-4614-7138-7, https://doi.org/10.1007/978-1-4614-7138-7
12. Kadlec, P., Gabrys, B., Strandt, S.: Data-driven soft sensors in the process industry. Computers & Chemical Engineering **33**(4), 795–814 (Apr 2009). https://doi.org/10.1016/j.compchemeng.2008.12.012, https://doi.org/10.1016/j.compchemeng.2008.12.012
13. Kowalczyk, A.: Support vector machines succinctly. Syncfusion Inc (2017)
14. Kramer, M.A.: Nonlinear principal component analysis using autoassociative neural networks. AIChE Journal **37**(2), 233–243 (Feb 1991). https://doi.org/10.1002/aic.690370209, https://doi.org/10.1002/aic.690370209

15. Kusiak, A., Song, Z.: Combustion efficiency optimization and virtual testing: A data-mining approach. IEEE Transactions on Industrial Informatics **2**(3), 176–184 (Aug 2006). https://doi.org/10.1109/tii.2006.873598, https://doi.org/10.1109/tii.2006.873598
16. Laubscher, R., Engelbrecht, Q., Marais, C., et al.: Application of machine learning algorithms in boiler plant root cause analysis: a case study on an industrial scale biomass unit co-firing sugarcane bagasse and furfural residue at excessive final steam temperatures. In: Proceedings of the Annual Congress-South African Sugar Technologists' Association. pp. 283–293. No. 91, South African Sugar Technologists' Association (2018)
17. Marcano-Cedeno, A., Quintanilla-Dominguez, J., Cortina-Januchs, M.G., Andina, D.: Feature selection using sequential forward selection and classification applying artificial metaplasticity neural network. In: IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society. IEEE (Nov 2010). https://doi.org/10.1109/iecon.2010.5675075, https://doi.org/10.1109/iecon.2010.5675075
18. MathWorks: Introduction to feature selection. https://www.mathworks.com/help/stats/feature-selection.html (2020 (accessed November 04, 2020))
19. MathWorks: Sequential feature selection. https://www.mathworks.com/help/stats/sequential-feature-selection.html (2020 (accessed November 04, 2020))
20. McIntyre, K.B.: A review of the common causes of boiler failure in the sugar industry. In: Proc. S. Afr. Sug. Technol. Ass. vol. 76, pp. 355–364 (2002)
21. Roy, R.K., Das, S.K., Panda, A.K., Mitra, A.: Analysis of superheater boiler tubes failed through non-linear heating. Procedia Engineering **86**, 926–932 (2014). https://doi.org/10.1016/j.proeng.2014.11.115, https://doi.org/10.1016/j.proeng.2014.11.115
22. Sheriff, M.Z., Botre, C., Mansouri, M., Nounou, H., Nounou, M., Karim, M.N.: Process monitoring using data-based fault detection techniques: Comparative studies. In: Fault Diagnosis and Detection. InTech (May 2017). https://doi.org/10.5772/67347, https://doi.org/10.5772/67347
23. SWIERCZ, M.: Application of PCA for early leak detection in a pipeline system of a steam boiler. PRZEGLAD ELEKTROTECHNICZNY **1**(10), 192–205 (Oct 2019). https://doi.org/10.15199/48.2019.10.43, https://doi.org/10.15199/48.2019.10.43
24. Teran, C.: Crude fingerprinting and predictive analytics. Petroleum Technology Quarterly **22**, 53–63 (2017)
25. Venkatasubramanian, V.: The promise of artificial intelligence in chemical engineering: Is it here, finally? AIChE Journal **65**(2), 466–478 (Dec 2018). https://doi.org/10.1002/aic.16489, https://doi.org/10.1002/aic.16489
26. Venkatasubramanian, V., Chan, K.: A neural network methodology for process fault diagnosis. AIChE Journal **35**(12), 1993–2002 (Dec 1989). https://doi.org/10.1002/aic.690351210, https://doi.org/10.1002/aic.690351210
27. Wright, D.J., Calitz, J.: Setting up for the 2020s addressing south africa's electricity crisis and getting ready for the next decade. http://hdl.handle.net/10204/11282 (2020 (accessed May 25, 2020))

Evaluation of heuristic guided character level word models for morphological segmentation of isiXhosa

Lulamile Mzamo[1][0000−0002−8867−7416], Albert Helberg[1][0000−0001−6833−5163], and Sonja Bosch[2][0000−0002−9800−5971]

[1] North-West University, Potchefstroom, South Africa,
lula_mzamo@yahoo.co.uk, albert.helberg@nwu.ac.za
[2] UNISA, Pretoria, South Africa,
boschse@unisa.ac.za

Abstract. The IsiXhosa Heuristics Maximum Likelihood Segmenter (XHMLS), an unsupervised isiXhosa text segmenter, is evaluated. XHMLS contributes new context free word probabilistic models for isiXhosa word generation and morphology heuristics as a guide to the models in the segmentation of isiXhosa words. Four guided models with an option for modified Kneser-Ney (mKN) smoothing are presented. XHMLS's boundary identification accuracy of 75.2 ± 0.18% underperformed when compared to the general benchmark Morfessor-Baseline's 77.2 ± 0.10%, and the current isiXhosa unsupervised segmentation benchmark, XBES's 77.44 ± 0.32%, but its accuracy outperforms both for small training set sizes and was over 70% across all the training set sizes. Two XHMLS models outperformed the F1 Scores of Morfessor-Baseline's, 48.9 ± 0.75%, and XBES's 58 ± 0.10% with F1 Scores of 59.4 ± 0.20% and 59.5 ± 0.30%.

Keywords: Natural language processing · Unsupervised machine learning · Morphological segmentation · Branching entropy · isiXhosa.

1 Introduction

Work on the unsupervised segmentation of isiXhosa text, using the IsiXhosa Heuristics Maximum Likelihood Segmenter (XHMLS), was presented in [32]. This paper details and evaluates XHMLS against the general benchmark Morfessor-Baseline and the current IsiXhosa unsupervised segmentation benchmark, isiXhosa Branching Entropy Segmenter (XBES) in terms of morpheme boundary identification accuracy and F1 Score.

The limited nature of human language resources and applications in South Africa calls for more work to be done in Human Language Technology (HLT). This lack can be attributed to the dependence on HLT expert knowledge, scarcity of annotated data resources, weak market demand for African languages, and how the particular language relates to other more resourced languages [40]. Morphological analysis, the task that XHMLS attempts to solve, is one of the basic tools in the natural language processing (NLP) of agglutinating languages. Morphological analysis is crucial in languages with rich systems of inflection and derivation. It is used as a pre-processing step to information retrieval, parsing, translation, word embedding and many other tasks [22, 23] .

IsiXhosa, one of the South African official languages belonging to the Nguni language family, is classified among the "resource scarce languages". The second largest language in South Africa with 9.3 million mother-tongue speakers (17% of the South African population), second only to isiZulu [45], isiXhosa has recently seen an increase in HLT tools, however, this increase has been from a low baseline [28].

The close morphological structure that isiXhosa has with other Nguni languages, i.e. isiZulu, Siswati and isiNdebele, avails this work for bootstrapping to these languages as has been shown in [6]. Nguni languages account for 45.8% of the South African mother tongue speaker population.

In this paper, we evaluate the surface segmentation performance of four conditional distribution models of isiXhosa word morphology guided by a set of isiXhosa word morphology heuristics. We use language modelling to estimate the probability distributions of the models and we test the effect of modified Kneser-Ney [8] smoothing on each.

This paper is structured as follows: Section 2 gives an overview of morphological segmentation in isiXhosa including information on the latest work in automated isiXhosa segmentation. Section 3 3 gives an overview of the current trends in unsupervised morphological segmentation in general including the current benchmarks. Section 4 presents the different models implemented and evaluated in this paper. Section 5 presents the experimentation and the results, and Section 6 concludes and recommends future work.

2 Morphological segmentation for isiXhosa

2.1 Morphological segmentation

Morphological analysis is the task of decomposing a word into its constituent units, the morphemes [25], and classifying them. Morphemes are the smallest meaning bearing component of a word [21]. In a language such as isiXhosa, meaningful units (morphemes) of the same linguistic word are joined in practical orthography. Therefore, a single word may represent a complete sentence.

It is important to note that there is a difference between morphological segmentation, which decomposes words into their constituent morphemes and morphological analysis which also classifies the identified morphemes [18, 52]. The task handled by XHMLS is morphological segmentation.

2.2 Morphological segmentation in isiXhosa

IsiXhosa is an agglutinating and polysynthetic language because its words are composed of a number of morphemes [21]. It is also fusional/inflectional because it exhibits some morpheme boundary fusion. Morpheme boundary fusion results from the fusion of two consecutive morphemes, and is difficult to distinguish in surface forms. An example is the *w* in *ukwanda* (to grow) which is linguistically segmented as *u-ku-and-a*. The *w* is a result of a fusion between the *u* and *a* vowels.

An isiXhosa word is composed of a root, a prefix, a suffix or a circumfix that attaches to the root. The root is the meaning carrying constituent of the word. A circumfix is the "simultaneous affixation of a prefix and suffix to a root to express a single meaning" [21]. An example of a circumfix in isiXhosa is the combination "a. . . ang.." in isiXhosa negation, e.g. ***a****-ka-hamb-****ang****-a (he/she did not go).*

Each of the affixes (i.e. prefix, suffix or circumfix) is composed of one or more affix morphemes. Morphemes follow one another in an order prescribed for each word type [24]. Most isiXhosa roots are bound morphemes, never appearing independently as words which are independently meaningful [36]. They at least appear as stems, which are word roots suffixed with a termination vowel [24], e.g. *anda* in *ukwanda*. Most isiXhosa morphemes belong to closed classes except for the roots, which belong to open classes. Closed classes are those morpheme classes where all the morphemes in a class are known, e.g. pronouns. Open morpheme classes are those that are subject to the productive nature of word formation.

2.3 Automated morphological segmentation of isiXhosa

One of the earliest works on automated morphological segmentation of South African languages, the automatic acquisition of a Directed Acyclic Graph (DAG) to model the two-level rules for morphological analysers and generators, is reported in [47]. The algorithm's accuracy was 100% for the inflection of isiXhosa noun locatives analysed.

More morphological analysis was done for other Nguni languages, including isiXhosa, by bootstrapping an existing isiZulu analyser [6, 37]. The study reported that in a small experiment with 181 isiXhosa words, 93.30% of the words were analysed.

Even more work on the development of text resources for South African languages was presented by (2014), including morphologically analysed corpora for ten languages including isiXhosa. Much of the isiXhosa morphological segmentation corpus, rated at an accuracy of 84.66%, is used as the test corpus in this study.

The most recent works for isiXhosa segmentation are those of [30] , which introduced a lemmatiser for isiXhosa, [34] presented the development of a rule-based noun stemmer for isiXhosa and the branching entropy morphological segmenter, the IsiXhosa Branching Entropy Segmenter (XBES), by { [31] . The isiXhosa lemmatiser was evaluated at an accuracy of 83.19%, the noun stemmer showed an accuracy rate of 91%, and XBES achieved an accuracy rate of 77.44% and an f1 score of 58%.

3 Unsupervised Morphological Segmentation

The last work done for morphological segmentation for isiXhosa reported in [31] uses unsupervised machine learning in the morphological segmentation of isiXhosa. This is attractive because it bypasses the need for expensive linguistic experts or annotation of training data.

3.1 Supervision in Machine Learning

There are four modes of training a machine learning model, namely supervised, semi-supervised, unsupervised [25] and reinforcement learning [29]. The difference between supervised and unsupervised learning depends on the character of the training data. If the data contains solution examples, the model is supervised. Data that contains solutions is also referred to as annotated data. Unsupervised learning learns from raw unannotated data. Semi-supervised learning is the spectrum between supervised and unsupervised learning. This could be by training the model from limited annotated data combined with large raw data or unannotated data with rules built into the model. In reinforcement learning the system is given feedback on its performance after several attempts at a solution. The system then learns from this feedback and adjusts its strategy as it attempts to build a model.

The segmenter evaluated in this paper, XHMLS, uses unsupervised learning in the segmentation of isiXhosa.

3.2 Unsupervised morphological segmentation works

The earliest works in unsupervised morphological segmentation used a form of accessor variety, where a morpheme boundary is identified by the possible number of letters that may follow that sequence of letters [11, 14]. This evolved to using mutual information [46, 51], and different forms of Branching Entropy [1, 46].

Minimum Description Length (MDL) [38] has seen extensive use, primarily as a measure of fit of the training data to heuristic models and statistical models [17, 20]. The comparative standard used in this study, Morfessor-Baseline [9], uses MDL and Maximum likelihood estimation.

Another popular approach is clustering and paradigmatic models. A paradigm is a grouping of words according to their form-meaning correspondence [5]. This involves clustering related words into a paradigm using a similarity measure, identifying the stem, and considering the rest as a sequences of affixes [7, 15]. The similarity measures used are Latent Semantic Analysis [10], Dice and Jaccard coefficients [25], affixality measurements [26] and Ordered Weighted Aggregator operators [7]. Word context is also another technique that is used to identify similar words [2].

Non-parametric Bayesian techniques have also shown promise, including Pitman-Yor process based models [48] and adaptor grammars [41]. These use Markov Chain Monte Carlo (MCMC) simulation with Gibbs Sampling [16] for inference.

Contrastive Estimation [33,42] is another non-parametric model that is showing elegance and promising results.

Language model based techniques have also been used. These tend to use heuristics and a language model to estimate the probabilities [13,35]. The work presented in this paper is one such endeavour.

It would be amiss to skip the performance of recent developments in artificial neural networks architectures even though they are not strictly unsupervised. Artificial Neural Network have started to push the performance of morphological segmentation. This is due to word and character-level embeddings [27] and [3]. The latest techniques depend on highly optimised big data embeddings [49,50]and Long Short Term Memory (LSTM) Neural Networks [19] driving conditional random field (CRF) emitters.

Several studies use a combination of the above techniques and measures [34, 39].

3.3 Choice of unsupervised segmenter for benchmarks

To place this work amongst other segmenters, a standard in morphological segmentation was chosen for comparison. The benchmark segmenter had to be publicly available and had to have been used for highly agglutinative languages like isiXhosa. The Morfessor-Baseline segmenter [9] was chosen because it has been used as a benchmark extensively and is freely available.

To establish a minimum performance baseline a random segmenter that randomly decides whether a point in a word is a boundary of a segment or not was implemented.

4 IsiXhosa Heuristics Maximum Likelihood Segmenter (XHMLS)

In this paper, we introduce the IsiXhosa Heuristics Maximum Likelihood Segmenter (XHMLS). XHMLS uses isiXhosa language heuristics to guide a probabilistic graphical model in the segmentation of isiXhosa words. The segmentation of the words is context free and only considers the probabilistic distribution of morphemes within the words. The heuristics generate the possible segmentations and the probabilistic graphical models rate the possible segmentations, and chooses the segmentation with the highest rating. This approach is similar to adaptor grammars [41], replacing the adaptor with modified Kneser-Ney smoothing and the grammar with heuristics.

In this work, we compare four proposed probabilistic graphical models for modelling isiXhosa word generation. The models are guided with isiXhosa heuristics to reduce the search field. We also test whether Kneser-Ney smoothing has any effect on the performance of the models.

For the models, different statistical data were kept. Where there is a sequence of morphemes, the modelling has an option of smoothing using the modified Kneser-Ney smoothing [8] which estimates the Hierarchical Pitman-Yor distribution. Information on the heuristics and the models evaluated is detailed below.

4.1 IsiXhosa Morphology Heuristics

We employ heuristics of the morphology of isiXhosa based on a rough summarisation of the language's morphology. These heuristics are not linguistic but are meant to simplify isiXhosa segmentation. For a start, prefix morphemes are syllabic, including when the word starts with a vowel [24].

Consider the noun *umntu* [person]. The base lemma for *umntu* is *ntu*, which consists of the root, *nt*, and the terminal vowel, *u*. We chose to over-segment by considering the terminal vowel and the pre-prefix as separate morphemes. So we segment *umntu* to *u-m-nt-u*. The prefix, *u-m*, is composed of $\underline{u}$, the pre-prefix, and *m*. The diminutive form of *umntu* is *umntwana* [child], which is linguistically segmented as *u-m-ntu-ana*. Because we are working on surface segmentation, which is a segmentation such that the morphemes can be concatenated to the unsegmented token, we segment *umntwana* to *u-m-nt-w-an-a*. This makes the suffix *w-an-a*. The difference here is that the *w* is now a morpheme of its own and the linguistic diminutive suffix *ana* is further split. This matters because the last *a* in *ana* changes when *umntwana* is changed to a locative, *emntwaneni* [to the child]. Linguistically *emntwaneni* is segmented as *e-m-ntu-ana-ini*. Production of a locative involves circumfixing a base word with *e..ini*. As indicated, the pre-prefix has been replaced by the locative prefix. Our surface segmentation for *emntwaneni* is *e-m-nt-w-an-en-i*. This kind of approach works well for verbs as well. The verb lemma *pheka* [cook] can be made into *uyandiphekela* [s/he is cooking for me] in isiXhosa, which can be segmented to *u-ya-ndi-phek-el-a*. Therefore, isiXhosa surface morphology can be summarised as:

- The first vowel is always a prefix morpheme (pre-prefix or other);
- The last vowel is always a suffix morpheme (terminal vowel);
- Prefix morphemes are complete syllables, except for '*m*' which has a silent vowel when followed by a consonant;
- Roots usually start with a consonant and end in a consonant;
- Suffix morphemes are reverse syllables, so they start with a vowel and end in a consonant, except for the terminal vowel and for '*w*' which has a silent preceding vowel when following a consonant.

During training, each training word is split into all possible combinations of the above heuristics and n-gram statistics are kept. Inference is done by choosing the segmentation that increases the probability of the word among all the possible segmentations generated by the heuristics for a word.

4.2 Probabilistic Language Models

Probabilistic graphical models are a diagrammatic representation of probability distributions (Bishop, 2006:p.360). These models allow for simple visualisation of the structure of the probabilistic model, a clear view of the properties of the model including conditional dependence properties etc. PGMs come in two forms, directed graphs and undirected graphs. In this paper, we use directed graphs. Directed graphs use nodes to indicate the random variables and directed edges to indicate dependence. The model provides for a plating system to model repetition. Shaded nodes are observed variables whilst the unshaded ones are latent variables that ought to be inferred by the model. In our formulation, we take liberties with the PGM diagram and denote probabilistic dependence using a solid line, generation using dashed lines and recursion using plating. We adjust the boundaries of the plating system to ensure proper scope.

In the following models, the symbols are standardised such that, w is a word, $\mathfrak{s}$ is a word segmentation, r is a word root, c is a circumfix, m is a morpheme, p is a prefix, s is a suffix, p_i is the i-th prefix morpheme, s_j is the j-th prefix morpheme, m_k is the k-th morpheme, I is the number of prefix morphemes, J is the number of suffix morphemes and K is the number of morphemes.

Axiom 1. **Vanishing sequences convention.** *Consider sequences* $\overrightarrow{x} = x_i, x_{i+1}, .., x_{i+n}$ *and* $\overleftarrow{x} = x_{i-n}, .., x_{i-1}, x_i$ *where* $n, i \in \mathbb{Z}$. *For convenience and as a convention in this paper, both sequences* $\overrightarrow{x}$ *and* $\overleftarrow{x}$ *reduce to* x_i *when* $n = 0$, *and vanish to an empty list, nothing, when* $n < 0$. *Furthermore,* $p\left(y|\overrightarrow{x}\right)$ *and* $p\left(y|\overleftarrow{x}\right)$ *reduce to* $p\left(y|x_1\right)$ *when* $n = 0$ *and* $p(y)$ *when* $n < 0$.

Definition 1. *The* ***truncation operator*** $\langle m_{1..k}\rangle_{[n}$ *on a sequence* $m_{1..k}$ *is defined as a function that returns a vanishing sequence such that variables.*

$$\langle m_{1..k}\rangle_{[n} = \begin{cases} \text{nothing}, & n \leq 0 \\ m_{1..k}, & n \geq k \\ m_{k-n+1}, .., m_{k-1}, m_k, & \text{otherwise} \end{cases} \tag{1}$$

The models that follow are based on our estimation of how the different morphemes relate, and they are new.

Morpheme Sequence Model (MS) The simplest model is the Morpheme Sequence (MS) model. The model proposes that an isiXhosa word is made up of a root and affixes. We consider the word root to be one of the morphemes and we model all the word morphemes to have a Markov sequential dependence of some order n.

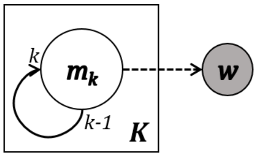


Fig. 1. Morpheme sequence PGM

In Fig. 1 we use the plate notation of PGM. The plate notation means that whatever is inside the plate is repeated by the number of times specified on the right bottom corner of the plate, in this case, K times. The probability of the word w when modelled by the segmentation $\int$ is estimated to be

$$P\left(w|\mathfrak{s}\right) = \prod_{k=1}^{K} P\left(m_k \middle| \langle m_{1..k-1}\rangle_{[n}\right), \tag{2}$$

where the Markov order of the dependencies is n.

Independent Affixes (IA). The Independent Affixes (IA) model is also one of the simplest models. The assumption of affix independence is also used by Morfessor-Baseline (Creutz & Lagus, 2002). It proposes that an isiXhosa word is made up of a root and affixes that are independent of each other, and that the affix morphemes within an affix have a Markov dependence of order n between them.

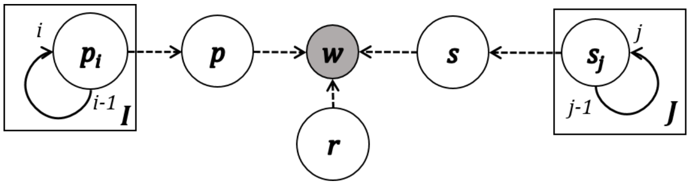


Fig. 2. Independent Affixes PGM

Of course, a word can have an empty prefix or an empty suffix. The probability of the word under the independent affixes model is estimated as

$$P\left(w|\mathfrak{s}\right) = P\left(r\right) \bullet \prod_{i=1}^{I} P\left(p_i \middle| \langle p_{1..i-1}\rangle_{[n}\right) \bullet \prod_{j=1}^{J} P\left(s_j \middle| \langle s_{1..j-1}\rangle_{[n}\right). \tag{3}$$

It is important to note that the relationship between p and p_i is that of generation, that p is made up of I p_i's. A similar relationship is established between s and s_j.

Affix-Links (AL). The Affix-Links (AL) model proposes that a word is made up of a root and affixes, and that the dependence between the root and the affixes is through the first prefix morpheme and the last suffix morpheme. If there are I prefix morphemes with the first morpheme being p_1 and there are J suffix morphemes with the last morpheme being s_J, then the prefix attaches to the root through the first prefix morpheme, p_1, and the suffix attaches to the root through the last suffix morpheme, s_J, and the rest of the affix morphemes attach to these linking morphemes and the root in a Markov Sequence of order n.

Fig 3 shows the Probabilistic Graphical Model for Affix Links. In this model, we don't show variables for the generated prefix and suffix as they don't feature

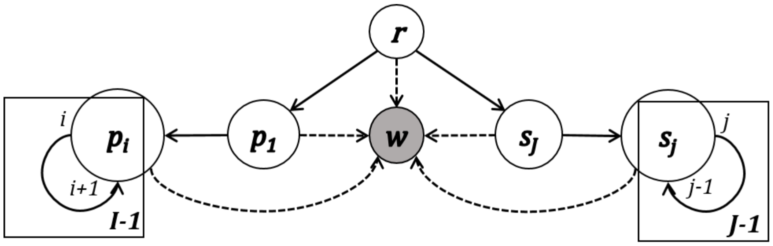


Fig. 3. Affix Links PGM

in the dependence structure. The probability formulation of the word under a segmentation is therefore

$$P\left(w|\mathfrak{s}\right) = P\left(r\right) \bullet \prod_{i=1}^{\mathbf{I}} P\left(p_i \middle| \left\langle r, p_{1..i-1} \right\rangle_{[n}\right) \bullet \prod_{j=1}^{\mathbf{J}} P\left(s_j \middle| \left\langle r, s_{\mathbf{J}..j+1} \right\rangle_{[n}\right). \qquad (4)$$

It is important to note that the sequence dependencies for the suffix are reversed, and the boundary of the plate model has been modified to only apply to the affix morpheme sequence dependence.

Circumfix Link (CL). This model proposes that in generating an isiXhosa word a root is chosen and inflected with a circumfix, which is composed of a prefix and a suffix. In addition, the circumfix used is dependent on the root. A circumfix can have an empty prefix thereby presenting as a suffix only and can have an empty suffix thereby presenting as a prefix only.

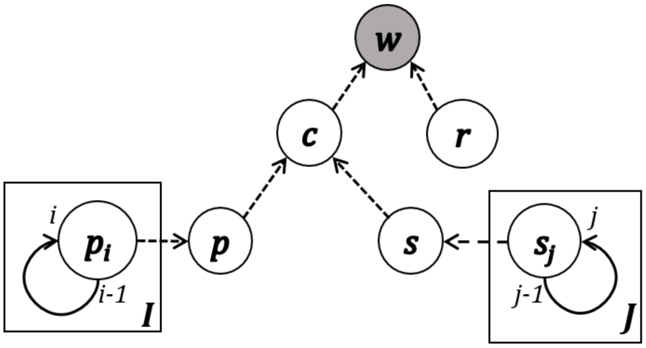


Fig. 4. Circumfix-Link PGMs

The model is shown in Fig. 4. When w is observed, D-Separation [4] creates a dependence between r and c, and a dependence between s and p leading to the probability formulation specified in (**5**)-(**8**).

$$P\left(w|\mathfrak{s}\right) = P(r|c) \bullet P(s|p) \bullet P(p) \tag{5}$$

where

$$P\left(c\right) = \prod_{\substack{k=1, \\ m_{1..K} = p_{1..I} \oplus s_{1..J}}}^{K} P\left(m_k \middle| \langle m_{1..k-1} \rangle_{[n}\right), \tag{6}$$

$$P\left(p\right) = \prod_{i=1}^{I} P\left(p_i \middle| \langle p_{1..i-1} \rangle_{[n}\right), \tag{7}$$

$$P\left(s\right) = \prod_{j=1}^{J} P\left(s_j \middle| \langle s_{1..j-1} \rangle_{[n}\right), \tag{8}$$

and $\oplus$ denotes the concatenation of two lists.

5 Evaluation

This section details the evaluation that was done on XHMLS. We give an indicated of the data sources used, how the data was split for evaluation purposes, the experi-ments setup and the results.

5.1 Data sources [1]

A raw unannotated isiXhosa corpus of 1.45 million isiXhosa words was compiled from the isiXhosa version of the South African Constitution [44], isiXhosa text on the internet and the IsiXhosa Genre Classification Corpus [43]. This text is named the training corpus.

For testing purposes, the NCHLT IsiXhosa Text Corpus (29 511 tokens) was used.

5.2 Data Splits

For training purposes, ten-fold training was performed for different training set sizes and language model n-grams lengths. The training set sizes chosen were multiples of ten (10) from one hundred (100) words to a million words and one and a half million (1.5 million) words.

[1] The IsiXhosa Genre Classification Corpus and NCHLT IsiXhosa Text Corpus are available at the South African Centre for Digital Language Resources (SADiLaR) (https://www.sadilar.org).

For testing purposes, a subset of the NCHLT corpus was used. Because the NCHLT corpus was generated with a rule based morphological analyser, the solutions are not all surface segmentations, others include linguistic morphemes. As an example the linguistic segmentation of *ukwanda* is *u-ku-and-a*. A surface segmenter, as XHMLS is, would segment *ukwanda* to *u-kw-and-a*. Excluding these kinds of entries resulted in an evaluation testing corpus of 13 441 tokens.

5.3 Experiment Setup

Training was performed for three segmenters, i.e. XHMLS, XBES and Morfessor-Baseline, using the training corpus, and tested against the testing corpus. The random segmenter does not require training. Morfessor-Baseline, XBES and XHMLS were trained with different training set sizes. Because Morfessor-Baseline does not support specifying n-gram size only XBES and XHMLS were trained to different model n-gram lengths.

Evaluation of the segmentations was measured as macro boundary identification accuracy and F1 Score, where, in a word, a morpheme boundary location is tagged 1 and everything else 0. Accuracy measures how many boundaries and non-boundaries the segmenter identified correctly. The F1 Score focuses on the possible boundary location and does not factor the non-boundary word locations.

5.4 Results

Starting with the benchmarks 10-fold validation results as show in Table 1, the benchmark accuracy is 77.4% from XBES which is comparable to Morfessor-Baseline's 77.2%. The random segmenter presents an average accuracy of 50.1%, as the minimum accuracy required. Any segmenter below this threshold actively degrades segmentation. The Random Segmenter's average F1 Score is 35.7% whilst Morfessor-Baseline's performance peaks at 48.9%, and XBES at 58.0%, making XBES the F1 Score benchmark. The results are shown with configuration information (i.e. smoothed or not, the training set size and the language model n-gram level that produced the results).

Table 1. Boundary Identification Results

Method	**Highest Accuracy (Smoothing/Op/training Size/n-gram length)**	**Highest f1 Score (Smoothing/Op/training Size/n-gram length)**
Random	50.1 ± 0.16	35.7 ± 0.16
Morfessor-Baseline	**77.2 ± 0.10(1m)**	48.9 ± 0.75 (10K)
Best XBES	**77.4 ± 0.32 (No/Sum/1.5m/11)**	58.0 ± 0.10 (No/Sum/1.5m/9)
XHMLS-MS	74.7 ± 0.24(Yes/1m/4)	**59.5 ± 0.30(Yes/1.5m/4)**
XHMLS-IA	75.2 ± 0.18(No/100K/2)	**59.4 ± 0.20(No/10K/3)**
XHMLS-AL	73.7 ± 0.01(Yes/1.5m/2)	59.0 ± 0.04(Yes/1.5m/3)
XHMLS-CL	74.0 ± 0.01(Yes/10K/4)	49.6 ± 0.04(Yes/10Km/4)

Fig. 5 and Fig. 6 show the respective accuracy and F1 Score trends trends relative to the training set size. Because the random segmenter is not trained, it is represented as a flat line across the training set sizes.

From Table 1, the best 10-fold average accuracy of XHML is, 75.2%, and is achieved by the Independent Affixes model (XHMLS-IA) at a training set size one hundred thousand (100 000) words using an unsmoothed bigram language model. This accuracy is considered inferior to both to Morfessor-baseline accuracy of 77.2% and XBES's 77.4% as the Wilcoxon Signed Rank test [12] p-value against the two was measured at 0.001953. The accuracy of the XHMLS with the Independent Affixes model outperforms both Morfessor-Baseline and XBES for training sets below one hundred thousand words (100 000) as shown in Fig. 5.

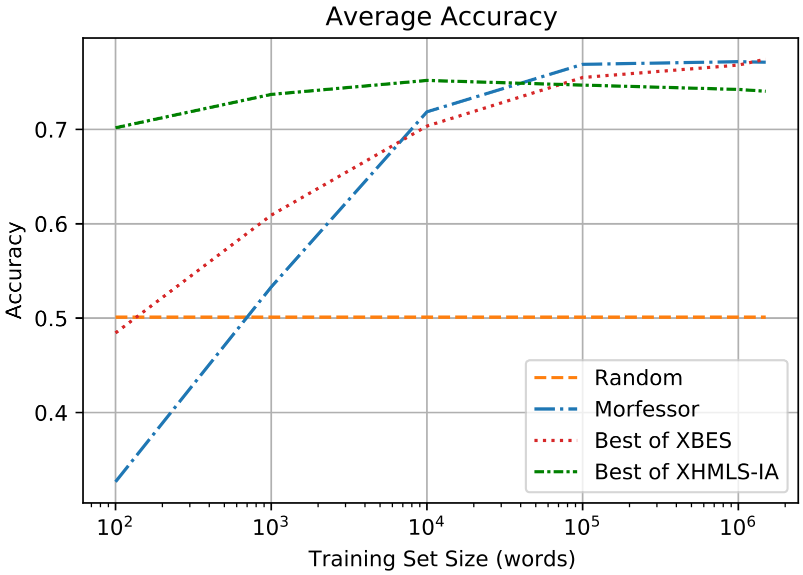


Fig. 5. Average accuracy of the segmenters

For the F1 Score, however, XHMLS has two modes that outperform the best XBES and Morfessor-baseline scores. They are the Morpheme Segments and the Independent Affixes modes at 59.5 % and 59.4 % respectively. The two are not statistically significant from each other as their p-value is 0.49219. They are, however, statistically significant from XBES's 58.0% as the p-value between XBES and the two is 0.001953. For the Morpheme Segments model, this is achieved using a smoothed 4-gram trained with 1.5 million words. For the Independent Affixes (XHMLS-IA) model this is achieved at ten thousand words with an unsmoothed trigram model. As is evident from Table 1 the effect

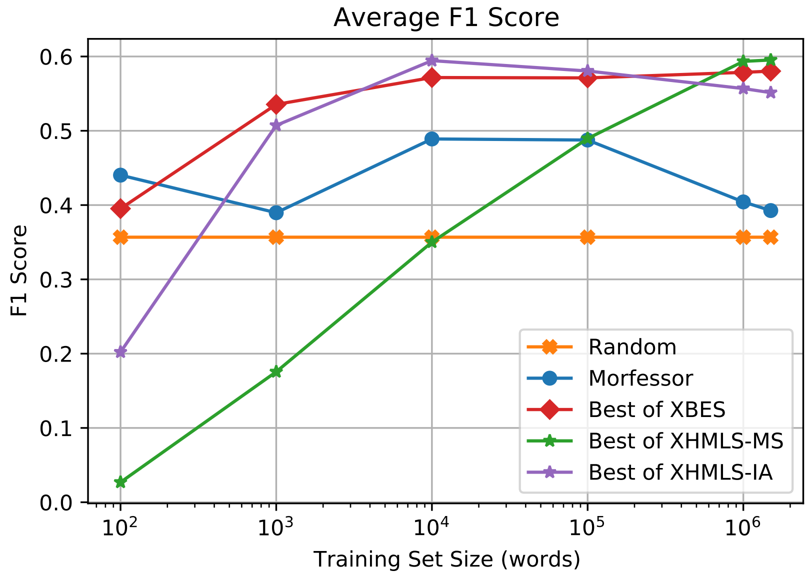


Fig. 6. Average F1 Score of the segmenters

of Kneser-Ney smoothing improves performance for all the models except the Independent Affixes model.

6 Conclusions

In this paper, an unsupervised morphological segmenter for isiXhosa that uses probabilistic language modelling is evaluated. The IsiXhosa Heuristic Maximum Likelihood Segmenter (XHMLS) uses four graphical models to segment isiXhosa.

The study contributes the use of isiXhosa word morphology heuristics as a guide to probabilistic graphical modelling of the segmentation of isiXhosa words.

XHMLS did not provide the best boundary identification accuracy compared to Morfessor-Baseline and XBES for isiXhosa overall but performed much better than both for small training set sizes, and maintained high accuracy across all the training set sizes.

Regarding the F1 Score, two models of XHMLS, the Morpheme Segments and the Independent Affixes, provided the best scores at 59.5% and 59.4% respectively. The former achieved this using a smoothed 4-gram language model trained on one and half million words and the latter's achievement was due to an unsmoothed trigram trained with ten thousand words.

For future work, the techniques employed in this paper will be used for unsupervised segmentation of other Nguni languages. There is scope to also establish heuristics and appropriate models for other South African languages. Further, XHML could be a pre-processing step to improve machine translation and web embeddings for South African languages.

Acknowledgment. The authors thank the South African Centre for Digital Language Resources (SADiLaR) (`https://www.sadilar.org`) for providing a central source of data and resources for South African Natural Language Processing work. The authors also with to thank the anonymous reviewers for their careful reading of the paper and their many insightful comments and suggestions. This has helped to improve and clarify this paper.

References

1. Ando, R.K., Lee, L.: Mostly-Unsupervised Statistical Segmentation of Japanese: Applications to Kanji. In: Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference (2000)
2. Belkin, M., Goldsmith, J.: Using eigenvectors of the bigram graph to infer morpheme identity. In: Proceedings of the ACL-02 workshop on Morphological and phonological learning. vol. 6, pp. 41–47. Association for Computational Linguistics, Philadelphia, USA (2002)
3. Bengio, Y.: Learning Deep Architectures for AI. Now Publishers Inc. (2009). https://doi.org/10.1561/2200000006
4. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer (2006)
5. Booij, G.: The grammar of words: an introduction to linguistic morphology. Oxford University Press, third edit edn. (2012)
6. Bosch, S., Pretorius, L., Fleisch, A.: Experimental Bootstrapping of Morphological Analysers for Nguni Languages. Nordic Journal of African Studies **17**(2), 66–88 (2008)
7. Chavula, C., Suleman, H.: Morphological Cluster Induction of Bantu Words Using a Weighted Similarity Measure. In: Proceedings of SAICSIT '17. p. 9. No. September 26–28, Thaba Nchu, South Africa (2017). https://doi.org/10.1145/3129416.3129453, `http://pubs.cs.uct.ac.za/archive/00001225/01/morphological-cluster-induction-camera.pdf`
8. Chen, S.F., Goodman, J.: An Empirical Study of Smoothing Techniques for Language Modeling. Tech. rep., Computer Science Group, Harvard University, Cambridge, Massachusetts (1998), `https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/tr-10-98.pdf`
9. Creutz, M., Lagus, K.: Unsupervised Discovery of Morphemes. In: Proceedings of the 6th Workshop of the ACL Special Interest Group in Computational Phonology. pp. 21–30. No. July, Philadelphia, USA (2002)
10. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by Latent Semantic Analysis. Journal of the American Society for Information Science Sep **41**(6), 391–407 (1990)
11. Déjean, H.: Morphemes as Necessary Concept for Structures Discovery from Untagged Corpora. In: D.M.W. Powers (ed.) NeMLaP3/CoNLL98 Workshop on Paradigms and Grounding in Natural Language Learning. pp. 295–299. ACL (1998)
12. Demšar, J.: Statistical Comparisons of Classifiers over Multiple Data Sets. The Journal of Machine Learning Research **7**, 1–30 (2006). https://doi.org/10.1016/j.jecp.2010.03.005

13. Erdmann, A., Khalifa, S., Oudah, M., Habash, N., Bouamor, H.: A Little Linguistics Goes a Long Way: Unsupervised Segmentation with Limited Language Specific Guidance. In: Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology. pp. 113–124. No. August 2, Association for Computational Linguistics, Florence, Italy (2019). https://doi.org/10.18653/v1/w19-4214
14. Feng, H., Chen, K., Deng, X., Zheng, W.: Accessor Variety Criteria for Chinese Word Extraction. Computational Linguistics **30**(1), 75–93 (2004), https://www.mitpressjournals.org/doi/pdfplus/10.1162/089120104773633394
15. Gaussier, E.: Unsupervised learning of derivational morphology from inflectional lexicons. In: Proceedings of ACL'99 Workshop: Unsupervised Learning in Natural Language Processing. pp. 24–30 (1999)
16. Geman, S., Geman, D.: Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE **6**(6) (1984), https://pdfs.semanticscholar.org/62c3/4c8a8d8b82a9c466c35cda5e4837c17d9ccb.pdf
17. Golénia, B., Spiegler, S., Flach, P.A.: Unsupervised morpheme discovery with UNGRADE. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). vol. 6241 LNCS, pp. 633–640. Springer, Berlin, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15754-7_76, http://www.cs.bris.ac.uk/Publications/Papers/2001221.pdf
18. Hammarström, H., Borin, L.: Unsupervised learning of morphology. Computational Linguistics **37**(2), 309–350 (2011)
19. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997), http://web.eecs.utk.edu/$\sim$itamar/courses/ECE-692/Bobby_paper1.pdf
20. Kit, C.: A Goodness Measure for Phrase Learning via Compression with the MDL Principle. In: Ivana Kruijff-Korbayova (ed.) Proceedings of the Third ESSLLI Student Session. pp. 175–187 (1998), https://pdfs.semanticscholar.org/120d/b0372be64b0c2a52ff836932d98937582674.pdf
21. Kosch, I.M.: Topics in Morphology in the African Language Context. Unisa Press, Pretoria (2006)
22. Kotze, G., Wolff, F.: Syllabification and parameter optimisation in Zulu to English machine translation. South African Computer Journal **57**(December), 1–23 (2015). https://doi.org/10.18489/sacj.v0i57.323
23. Kudo, T.: Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates. arXiv prePrint arXiv:1804.10959v1 (2018)
24. Louw, J., Finlayson, R., Satyo, S.: Xhosa Guide 3 for XHA100-F. University of South Africa, Pretoria (1984)
25. Manning, C.D., Schütze, H.: Foundations of Statistical Natural Language Processing, vol. 26. MIT Press (1999). https://doi.org/10.1162/coli.2000.26.2.277
26. Méndez-Cruz, C.F., Medina-Urrea, A., Sierra, G.: Unsupervised morphological segmentation based on affixality measurements (2016)
27. Mikolov, T., Corrado, G., Chen, K., Dean, J.: Efficient Estimation of Word Representations in Vector Space. Proceedings of the International Conference on Learning Representations (ICLR 2013) pp. 1–12 (2013). https://doi.org/10.1162/153244303322533223, http://arxiv.org/pdf/1301.3781v3.pdf
28. Moors, C., Calteaux, K., Wilken, I., Gumede, T.: Human language technology audit 2018: Analysing the development trends in resource availability in all South

African languages. In: SAICSIT 2018. pp. 296–304. No. 26-28 September, ACM, Port Elizabeth, South Africa (2018). https://doi.org/10.1145/3278681.3278716

29. Murphy, K.P.: Machine Learning: A probabilistic perspective. MIT Press, Cambridge, MA,USA (2012)
30. Mzamo, L., Helberg, A., Bosch, S.: Introducing XGL-a lexicalised probabilistic graphical lemmatiser for isiXhosa. In: Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech) 2015. pp. 142–147. IEEE, Port Elizabeth, South Africa (2015)
31. Mzamo, L., Helberg, A., Bosch, S.: Evaluation of combined bi-directional branching entropy language models for morphological segmentation of isiXhosa. In: Barnard, E., Davel, M. (eds.) Proceedings of the South African Forum for Artificial Intelligence Research. pp. 77–89. No. 3-6 December, Cape Town, South Africa (2019)
32. Mzamo, L., Helberg, A., Bosch, S.: Towards an unsupervised morphological segmenter for isiXhosa. In: Proceeding of 2019 SAUPEC/RobMech/PRASA Conference. pp. 166–170. No. January 28-30, Bloemfontein, South Africa (2019)
33. Narasimhan, K., Barzilay, R., Jaakkola, T.: An Unsupervised Method for Uncovering Morphological Chains. Transactions of the Association for Computational Linguistics **3**, 157–167 (2015), https://github.com/
34. Nogwina, M.: Development of a Stemmer for the IsiXhosa Language. M.sc. dissertation, University of Fort Hare: MSc. Dissertation (2016), http://libdspace.ufh.ac.za/bitstream/handle/20.500.11837/221/MSc%28ComputerScience%29-NOGWINA%2CM.pdf?sequence=1&isAllowed=y
35. Nowakowski, K., Ptaszynski, M., Masui, F.: MiNgMatch-A fast n-gram model for word segmentation of the Ainu language. Information (Switzerland) **10**(317) (2019). https://doi.org/10.3390/info10100317
36. Pahl, H.: IsiXhosa. Educum Publishers, King Williams Town (1982)
37. Pretorius, L., Bosch, S.E.: Finite-State Computational Morphology: An Analyzer Prototype For Zulu. Machine Translation **18**(3), 195–216 (jul 2005). https://doi.org/10.1007/s10590-004-2477-4
38. Rissanen, J.: Modelling by the shortest data description. Automatica **14**, 465–471 (1978)
39. Schone, P., Jurafsky, D.: Knowledge-Free Induction of Morphology Using Latent Semantic Analysis. In: Proceedings of CoNLL-2000 and LLL-2000. pp. 67–72. Lisbon, Portugal (2000)
40. Sharma Grover, A., van Huyssteen, G.B., Pretorius, M.W.: South African human language technologies audit. Language Resources and Evaluation **45**(3), 271–288 (jun 2011). https://doi.org/10.1007/s10579-011-9151-2, http://link.springer.com/10.1007/s10579-011-9151-2
41. Sirts, K., Goldwater, S.: Minimally-Supervised Morphological Segmentation using Adaptor Grammars. Transactions of the Association for Computational Linguistics **1**, 255–266 (2013), http://www.aclweb.org/anthology/Q/Q13/Q13-1021.pdf
42. Smith, N.A., Eisner, J.: Contrastive Estimation: Training Log-Linear Models on Unlabeled Data. In: Proceedings of the 43rd Annual Meeting of the ACL. pp. 354–362. No. June, Ann Arbor, Michigan, USA (2005), http://www.anthology.aclweb.org/P/P05/P05-1044.pdf
43. Snyman, D., Van Huyssteen, G.B., Daelemans, W.: Cross-Lingual Genre Classification for Closely Related Languages. In: Twenty-Third Annual Symposium of the Pattern Recognition Association of South Africa. pp. 132–137 (2012), http://www.let.rug.nl/$\sim$vannoord/Lassy/deliverable1-1.pdf

44. South African Parliament: UMgaqo-siseko weRiphablikhi yoMzantsi-Afrika ka-1996 (1996), http://www.justice.gov.za/legislation/constitution/SAConstitution-web-xho.pdf
45. Statistics South Africa: Community survey 2016 in Brief (2016), https://www.statssa.gov.za/publications/03-01-06/03-01-062016.pdf
46. Sun, M., Shen, D., Tsou, B.K.: Chinese Word Segmentation without Using Lexicon and Hand-crafted Training Data. In: Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 2. pp. 1265–1271. No. August 10, Association for Computational Linguistics (1998), https://aclanthology.info/pdf/C/C98/C98-2201.pdf
47. Theron, P., Cloete, I.: Automatic acquisition of two-level morphological rules. In: Proceedings of the fifth conference on Applied Natural Langauge Processing. pp. 103–110. Morgan Kaufmann Publishers, San Francisco, CA, Washington, DC (1997)
48. Uchiumi, K., Tsukahara, H., Mochihashi, D.: Inducing Word and Part-of-Speech with Pitman-Yor Hidden Semi-Markov Models. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing. pp. 1774–1782. No. July 26-31, Association for Computational Linguistics, Beijing, China (2015), http://www.aclweb.org/anthology/P15-1171
49. Yan, H., Qiu, X., Huang, X.: A Graph-based Model for Joint Chinese Word Segmentation and Dependency Parsing. Transactions of the Association for Computational Linguistics **8**, 78–92 (2020). https://doi.org/10.1162/tacl, https://doi.org/10.1162/tacl
50. Yang, H.: BERT Meets Chinese Word Segmentation. arXiv prePrint arXiv:1909.09292v1 (2019)
51. Ye, Y., Wu, Q., Li, Y., Chow, K.P., Hui, L.C.K., Yiu, S.M.: Unknown Chinese word extraction based on variety of overlapping strings. Information Processing and Management **49**(2), 497–512 (2013). https://doi.org/10.1016/j.ipm.2012.09.004, http://dx.doi.org/10.1016/j.ipm.2012.09.004
52. Zwicky, F.: Entdecken, erfinden, forschen: im morphologischen Weltbild. Muenchen: Droemer (1966)

Future frame prediction in transformation space using goodSTN*

Nirvana Pillay$^{1,2[0000-0003-4999-1215]}$ and Edgar Jembere$^{1,2[0000-0003-1776-1925]}$

[1] School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, Westville Campus, Private Bag X54001, Durban 4000, South Africa
nirvanap02@gmail.com
jemberee@ukzn.ac.za
[2] Centre for AI Research (CAIR), South Africa

Abstract. The generation of future frames of a video involves the analysis of the previous i frames and the subsequent prediction of the following j frames. The current state-of-the-art solutions are able to generate a single frame with a high degree of photorealism. When the task is extended to generating a number of frames, the degree of photorealism decreases in that the predictions become blurry and motion is underrepresented. This is partially attributed, by existing research, to the prediction of future frames from pixel values. In this work, we generate future frames from predicted affine transformations using an architecture constituted of a stacked Convolutional LSTM (ConvLSTM) that parameterizes a good Spatial Transformer Network (goodSTN). An STN, enables the extraction and prediction of transformations by sampling a parameterized grid over the entire image. Our work investigated the improvement of computational time and possibly quality of results by calculating the transformations for only the Region of Interest using a goodSTN. We hypothesize that modelling motion as such, and generating frames from the predicted transformations results in a better representation of motion as well as more effective utilization of memory. Our proposed solution (ConvLSTM-goodSTN) was evaluated and benchmarked on the ConvLSTM-STN architecture, in terms of photorealism. The ConvLSTM-STN performed marginally better with a PSNR of 29 and a SSIM of 0.89 as compared to PSNR of 28 and a SSIM of 0.88. The former model, however, needed to be trained for 70 epochs whilst the latter only required 40.

Keywords: STN · ConvLSTM · Region of Interest · Frame prediction.

1 Introduction

A machine learning model that is able to generate a sequence of future frames of a video from a sequence of past frames is said to have learned an understanding of the physical environment and the internal representation of objects in a frame.

* The support of the Centre for High Performance Computing (CHPC) is gratefully acknowledged.

The application of such a model is in autonomous decision making areas such as self-driving cars, social robots [5] and video completion [7]. For example, a SocialGAN [5] determines plausible and socially acceptable walking trajectories of people, thus, aiding in the navigation of human-centric environments.

A sequential model that represents high dimensional data (i.e. images through time) as well as the inherent uncertainty of the future is required. In recent work, future frames were generated from predicted transformations. These models, extract the transformations between adjacent input frames, predict a future transformation, and apply it to the last frame of the input to generate the next frame and so forth. There is, therefore, no need to store low level details of the input as only the source of variability is modelled. The resultant frames are furthermore photorealistic as the integrity of objects are preserved through the transformation of their edges. An STN is an example of a network that enables the extraction and prediction of transformations. A traditional STN determines the affine transformation (i.e. translation, rotation and scaling) matrices over the entire image. As such the static areas of the video are considered when learning transformation values. In order to address the aforementioned issue, this work presents a goodSTN.

The goodSTN determines a Region of Interest (ROI) over which transformation matrices are calculated; the ROI is the immediate area that contains all the ‘good’ features selected by the *Shi-Tomasi Corner Detection* algorithm [12]. The goodSTN in conjunction with a ConvLSTM was implemented to predict several frames into the future.

The main contribution of this work is the design of the goodSTN which performs a crude object detection and, thereafter, calculates the transformations over the ROI. The range of applicability of this model is, therefore, limited to objects moving against a homogenous, static background. The ConvLSTM-goodSTN was compared to that of the ConvLSTM-STN and simply copying the last frame of the input sequence. These were evaluated on the basis of computational time and the quality of results generated (measured by means of PSNR, MSE and SSIM score).

The paper is organized as follows: Section 2 summarizes related work, Section 3 presents details of the implemented models, Section 4 discusses the experimental design and thereafter the results are presented in Section 5 and conclusions in Section 6.

2 Related Work

In the context of generative Machine Learning, a multitude of problems have been addressed including the prediction and generation of future frames of a video. Such a task requires the modelling of temporal data whilst preserving spatial information and accounting for the inherent uncertainty of the future.

In an attempt to model spatio-temporal relationships in video data (state-of-the-art) models include either CNNs, RNNs or both. The decisive criteria for network selection is its ability to model high dimensional sequential data accurately and its computational efficiency. For sequential modelling, RNNs are typically implemented due to their ability to capture temporal dependencies. However, the efficacy of CNNs in Computer Vision tasks has led to the implementation of a new class of CNNs to perform the same task, Temporal Convolutional Networks (TCNs).

A TCN in conjunction with a dilated CNN to model temporal and spatial dependencies respectively was implemented by [15]. A PGGAN architecture with a 3D CNN generator was similarly used by [1], thereby, preserving the spatial dependencies of the video. Another attempt at sequential modelling utilizing CNNs [2] was an architecture in which a network was replicated through time i.e. the output of the CNN at time t was the input to the CNN at time $t+1$. The resultant model was a 'peculiar RNN' as parameters were now shared across time whilst still convolving spatial data.

The effectiveness of RNNs at capturing temporal dependencies is owing to this parameter sharing across time. The inability to store historical data, however, results in vanishing gradients and, therefore, the inadequate capture of long-term dependencies. LSTMs and GRUs are both classes of RNNs with a memory cell to counteract this issue. A CNN-LSTM architecture was implemented by [9] to predict future frames of synthetic video data. These aspects were later united by [13] into a single network, a convolutional LSTM (ConvLSTM). The input to such a network is compressed into a 3D tensor that preserves spatial information. ConvLSTMs, therefore, possess strong representational power and are able to effectively make predictions on complex, dynamic systems. A stacked ConvLSTM to address the problem of future frame prediction as well as determine the state of motion of a robot arm was implemented by [3].

During training, some generative models attempt to optimize a pixel-wise loss function [2, 3] which results in an easier training procedure. These models, however, do not adequately account for the uncertainty of the future. The generated frame is an average of several closely related future frames which fails to represent finer features of the objects within the frame resulting in the blurring (or disappearance entirely) of these objects. Variational Autoencoders (VAEs) are built on functions such as MSE loss and are able to generate diverse futures due to their stochastic nature. The quality of the generated frames, however, still low due to the minimization of MSE loss [8]. GANs [4] do not suffer the same shortcomings as two adversary networks (the generator and the discriminator) compete in a minimax game and the optimal solution is found at Nash Equilibrium. GANs, however, are highly susceptible to mode collapse (the generator learns modes that will deceive the discriminator and restricts its outputs to these modes). A further difficulty with training GANs is that the convergence of both networks is not easy to attain.

The aforementioned models are differentiated by which aspects of the input data is analysed; namely the operation in either pixel space or transformation space. In pixel space, the model [1, 5, 9] generates future frames directly from pixel intensities, requiring the storage of pixel values of the input and, therefore, redundant information. The storage of redundant pixel level information results in the underestimation of motion of objects in the frame. In order to counteract this issue, [2,3,8,15] implemented models that modelled the source of variability and generated frames from predicted transforms thereby utilizing storage more effectively. Such a model applies predicted transformations to the last frame of the input resulting in the next frame of the sequence. This generation mechanism is further hypothesized [2] to counteract some of the blurriness associated with pixel-wise loss functions.

In order to effectively extract and predict transformations, [15] trained a CNN to calculate predicted transformations and, thereafter, determine the values of the output pixels using interpolation. A different mechanism was implemented by [2] in which transformation matrices are calculated over patches of the image by minimizing the mean squared error between the real and predicted image patch. Similarly, [3] used a Spatial Transformer Network (STN) [6] which created a grid over the entire image and parameterized the grid with affine transformations.

3 Modelling

A generative model (Figure 1) was designed to generate a sequence of 10 future frames (based on 5 context frames) in transformation space. As such the model constituted of stacked convolutional LSTMs (ConvLSTM) and a modified Spatial Transformer Network (STN), the goodSTN. Thus, the spatial aspects of the data were preserved whilst its complex and dynamic temporal aspects were modelled.

The generative model was based on the model proposed in [3]. However, our model receives no additional input such as the action and state of objects in the frame and our generator makes use of the goodSTN.

3.1 Generator in Transformation Space

A ConvLSTM is an LSTM with convolutional operations on both the input-to-state and state-to-state transitions and may be stacked to form an encoding-forecasting architecture that effectively models spatio-temporal data. The encoding module compresses input into a 3D hidden state tensor whilst the forecasting module predicts the future by unfolding this hidden state. The resultant network possesses strong representational power and is suitable to model complex, dynamic systems [13].

Our model (based on the model implemented in [3]) comprises of a ‘core trunk’, a one stride-2 5×5 convolution, and seven stacked ConvLSTM layers (the output of layer n is the input to layer $n + 1$). The weights of these layers are arranged into 5×5 convolutions. The input to the generator is a 5D tensor which contains information pertaining to a sequence of greyscale context frames i.e. the number of context frames, number of channels, batch size, image height and width. The resolution is reduced by size 2 down-sampling prior to layers 3 and 5 and, thereafter, upsampled by 2 after layer 5, 6, 7 and in the final convolutional layer. A transpose convolutional layer performs upsampling which ensures that coarse, salient features of an image are represented in a dense, detailed output. The final convolutional layer, thus, generates a full-resolution composite mask as output.

The output of ConvLSTM 5 is received by a Fully-Connected Layer which along with the ROI Selector and Grid Generator constitutes the goodSTN as per Section 3.2. The goodSTN generates transformation parameters i.e. 10 3×2 predicted affine transformation matrices; the predictions are for separate aspects within the frame. Therefore, the model also predicts a compositing mask over each transformation which is output by the final convolutional layer. The generated frame is reconstructed by applying predicted affine transformations, merged by masking, to the last input frame. For our purposes, such an architecture is agreeable as the most basic applications, such as future frame prediction for MNIST, require 2 stacked ConvLSTM [13]. The addition of another layer would not increase the quality of the generated frames.

The model is trained to minimize the mean square error loss between the sequence of generated frames and the sequence of ground truth frames.

3.2 Spatial Transformer Network (STN)

An STN determines the affine transformation matrix (i.e. translation, rotation and scaling) between adjacent frames and is traditionally comprised of the following components: localization network, grid generator and sampler. The localization network is a Fully-Connected network that receives a feature map, $U = [W, H, C]$, and outputs the transformation parameters θ. The grid generator, thereafter, creates a parameterized sampling grid which represents a set of points that should be sampled to produce the target co-ordinates of the output feature map. The output feature map, $V = [W', H', C']$, is then obtained via bilinear sampling. In this network, the entire image is considered when learning transformation values, including the static areas of the video.

To reduce this redundancy, the goodSTN was implemented in which the transformation matrices were calculated for only the Region of Interest (ROI), determined by means of a crude object detection as shown in Figure 2. During training, samples within a mini-batch are adjacent frames in a video and, thus, each mini-batch exhibits consistent motion. The ROI is, therefore, determined over the entire mini-batch to ensure that areas of motion are not excluded and that significant transformations are accounted for. The object detection involved the selection of 'good features' in a frame by means of the *Shi-Tomasi Corner Detection* algorithm. The algorithm quantizes the frame into windows and each window is classified as flat, edge or corner based on variations in image intensities. As a result, good feature points (occlusions, disocclusions and features corresponding to points in reality) for a sequence of frames over the entire batch are detected.

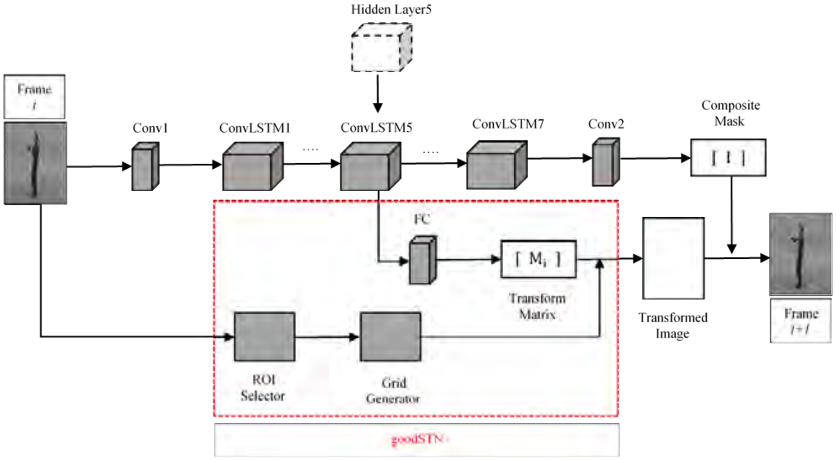


Fig. 1. Schematic of Generator

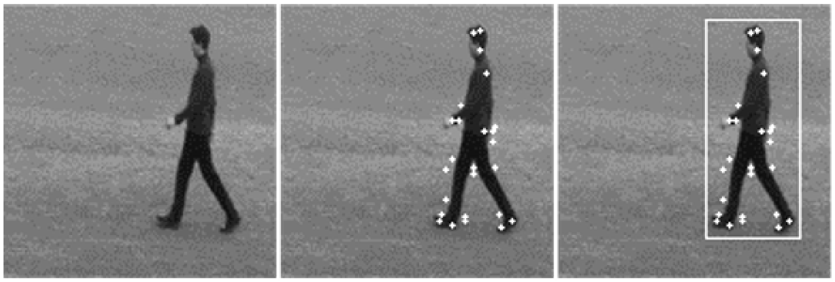


Fig. 2. ROI selection based on 'good' features in the frame

4 Experimental Design

The models, both the ConvLSTM-goodSTN and ConvLSTM-STN, were implemented in PyTorch. The optimal learning rate was determined using the method outlined by [14] which involved increasing the learning rate over each mini-batch and, thereafter, plotting it against the loss. The optimal learning rate was found at one order of magnitude less than that of the turning point of plot. A learning rate of 5×10^{-4} was selected for both the ConvLSTM-goodSTN and ConvLSTM-STN. The ADAM optimizer was used with the suggested hyperparameters of $\beta_1 = 0.9$ and $\beta_1 = 0.999$ [10].

The experiments were conducted on the KTH Action dataset [11]; a dataset that has been popularized as a means to train and evaluate action recognition models and, therefore, exhibits sufficient motion. The dataset comprises of 25 subjects performing 6 action types, namely walking, jogging, running, boxing, hand waving and hand clapping. The videos (with 25 fps) were pre-processed into its constituent frames and resized to 64×64 pixels. As per the experimental procedure of [1], 16 persons were used for training whilst the remainder was held-out for testing corresponding to a train/test split of 65/35.

The model received 5 frames as input and was tasked to predict and generate the next 10. The generated frames would be evaluated on photorealism and motion representation. An image, or consecutive generated frames, is deemed photorealistic if it depicts the realism of a real world image i.e. motion (the net change in the position of moving objects), object structure and texture is preserved. The generated frames were evaluated quantitatively via the following metrics: Peak Signal to Noise Ratio (PSNR), Structural Similarity Index (SSIM) and Mean Squared Error (MSE). PSNR is commonly used to quantify the quality of an image (degraded by compression) to the original. It is a function of the mean squared error between the two images and, therefore, does not account for perceptual differences as the SSIM does.

In the context of future frame prediction, these metrics compare generated frames to that of the ground truth counterpart. The above evaluation metrics were calculated for the generated sequences as well as a baseline sequence in which the future frames are the last frame of the sequence copied (‘*CopyLast*’). Such an evaluation does not account for frames that are photorealistic and plausible but differ from the ground truth. The frames, therefore, were qualitatively compared (presented in Section 5). A further evaluation criteria was the efficiency of each model in terms of the training time required.

5 Results and Discussion

Table 1 represents the average performance for each model compared to that of the baseline which are competitive with literature [1, 3]. The ConvLSTM-STN performs marginally better several frames into the future than the ConvLSTM-goodSTN at the expense of computation time (Table 2). In Figure 3, the training behavior for each model is presented (training of the ConvLSTM-goodSTN model used pretrained weights from a previous run, starting at Epoch 29). From the curves, it is evident that both models follow a similar training trajectory and experience significant fluctuations throughout. The training of the models were considered complete at the inflection point i.e. time step 13 (Epoch 40) for the ConvLSTM-goodSTN.

Table 1. Average performance across 10 frames for test set

	ConvLSTM-goodSTN	ConvLSTM-STN	CopyLast
MSE	0.0020	0.0018	0.0032
PSNR	28.579	29.104	28.0796
SSIM	0.8803	0.8922	0.8711

As more frames are generated further into the future, the quality degrades (as seen in Figure 4 to Figure 6) due to the model generating frames from previously

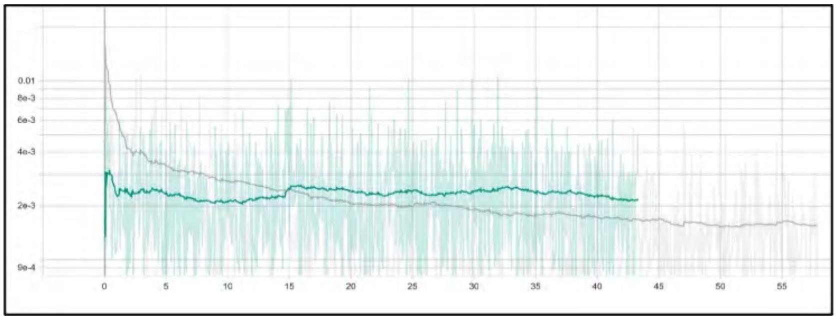


Fig. 3. Training behavior of models

Table 2. Computational parameters of each model

	ConvLSTM-goodSTN	ConvLSTM-STN
Number Parameters	7 701 086	7 701 086
Training Epochs	40	70
Training Time (hrs)	47	59

generated frames. The '*CopyLast*' baseline performs poorly on all metrics which implies that the models are indeed creating motion in the generated frames and compare better to the ground truth frames.

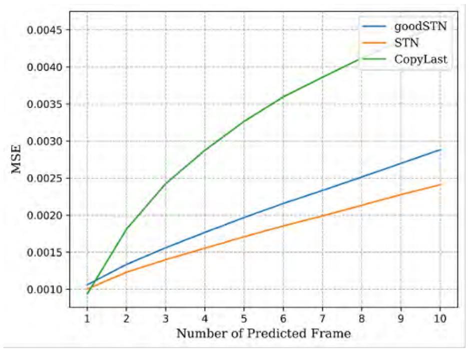


Fig. 4. Average MSE for each time step across the test set

In a qualitative comparison, both models perform approximately the same. It is further evident that the models generate plausible future frames. For example in Figure 7, the future position of the arm may extend upwards or retract towards the body. The frames, however, significantly blur over time because the optimization of MSE loss averages several closely related futures.

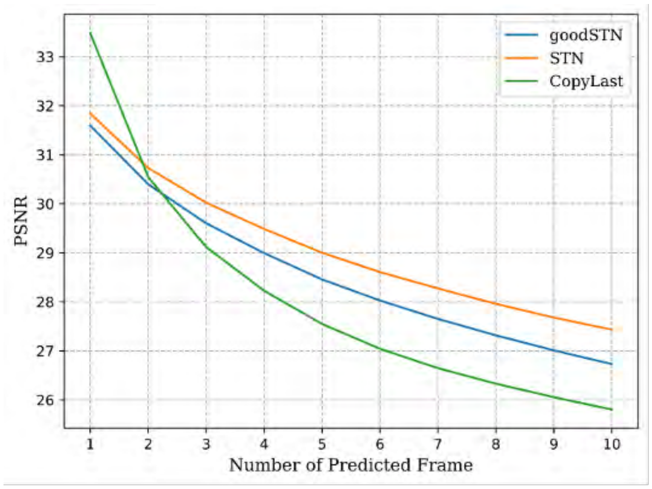


Fig. 5. Average PSNR for each time step across the test set

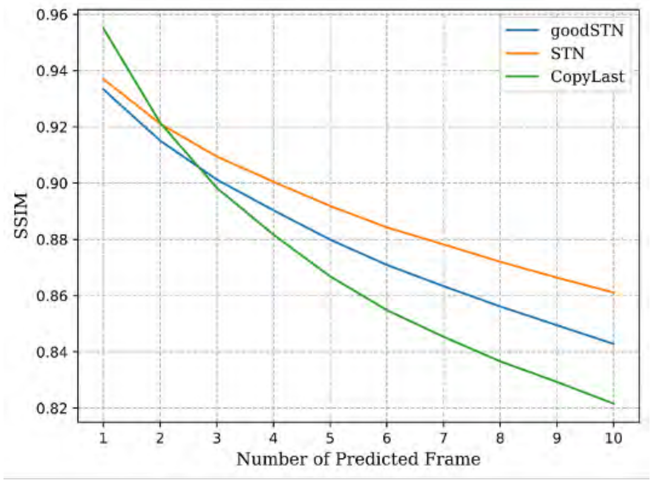


Fig. 6. Average SSIM for each time step across the test set

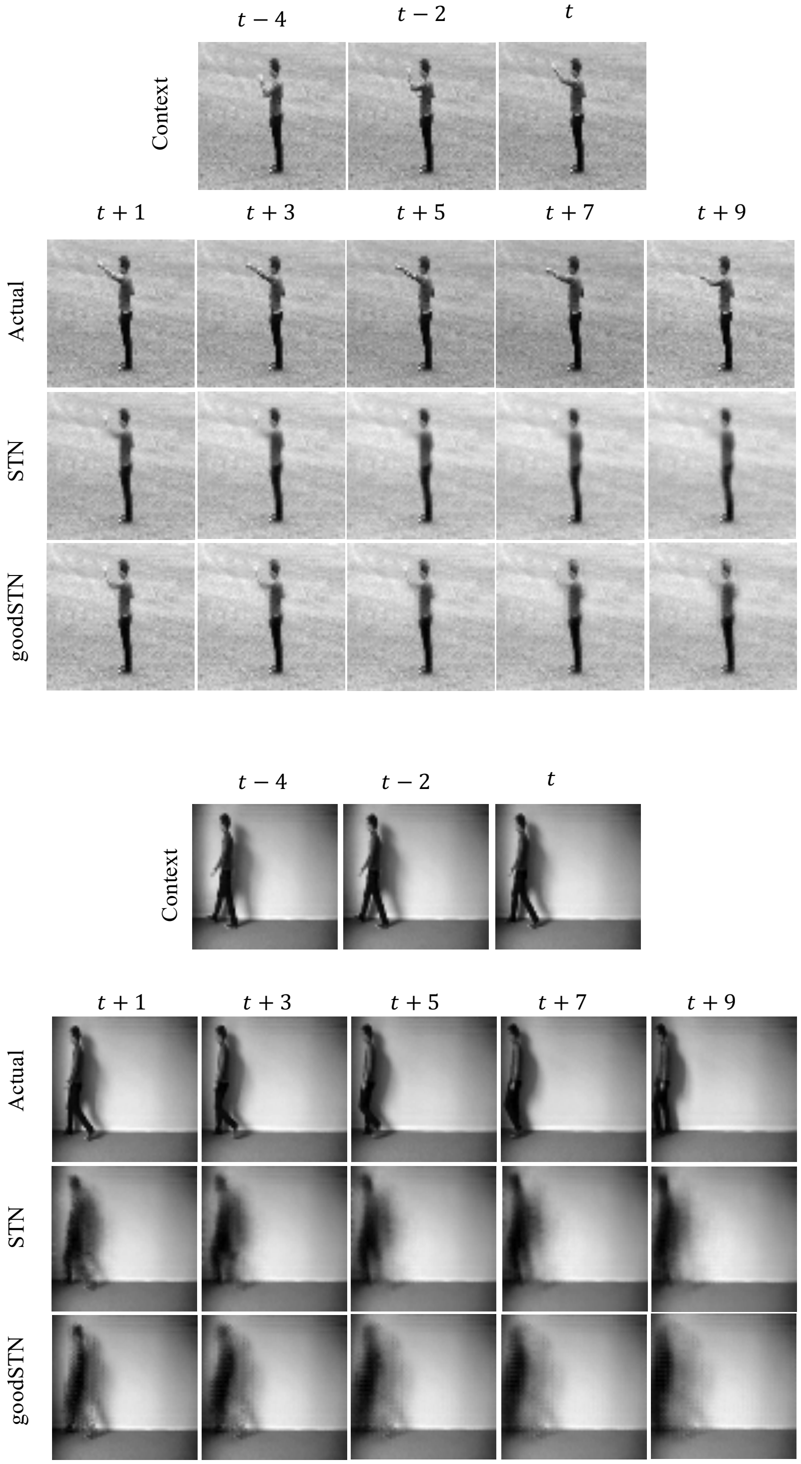


Fig. 7. Frames generated by each model and its ground truth counterpart

6 Conclusions and Future Work

The prediction and generation of future frames of a video is a complex task as it requires sequential modelling of images. Machine learning models that perform such a task are evaluated on photorealism, motion representation and the plausibility of the frames. The generation of frames from predicted affine transforms was evaluated for two models (ConvLSTM-goodSTN and ConvLSTM-STN); where the goodSTN calculated transformations for only the ROI. The results were evaluated quantitatively and the models performed approximately the same but the goodSTN enabled a shorter training time.

A qualitative evaluation, further, showed that the images degraded over time and the images blurred for both models. It is also evident that quantitative metrics are only a partial reflection on the quality of the generated frames. The blurring is due the optimization of the MSE loss function as it averages several closely related futures. It is, therefore important to consider alternative loss functions in future work and improve on the evaluation metrics.

Generative Adversarial Networks (GANs) [4] are capable of mitigating blurriness that results from the optimization of MSE loss. The constituent components of a GAN are the generator and discriminator which are engaged in a minimax game and the optimal solution is found at the Nash equilibrium. A Conditional GAN (CGAN) in transformation space will be implemented to generate plausible, photorealistic frames that adequately represent motion.

A more in-depth qualitative analysis will be implemented in the form of a modified Turing test namely a Two Alternative Forced Choice Test (2AFC). Such a test involves presenting a human subject with two video sequences and asking which sequence is preferable.

References

1. Aigner, S., Körner, M.: Futuregan: Anticipating the future frames of video sequences using spatio-temporal 3d convolutions in progressively growing gans. In: Photogrammetric Image Analysis & Munich Remote Sensing Symposium. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XLII-2/W16 (2019)
2. Amersfoort, J., Kannan, A., Ranzato, M., Szlam, A., Tran, D., Chintala, S.: Trans-formation-based models of video sequences.arxivpreprint (2017). https://doi.org/arXiv:1701.08435
3. Finn, C., Goodfellow, I., Levine, S.: Unsupervised learning for physical interaction through video prediction. In: Advances in neural information processing systems (2016)
4. Goodfellow: Nips 2016 tutorial: Generative adversarial networks (2016)
5. Gupta, A., Johnson, J., Fei-Fei, L., Savarese, S., Alahi, A.: Social gan: Socially acceptable trajectories with generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
6. Jaderberg, M., Simonyan, K., Zisserman, A., kavukcuoglu, k.: Spatial transformer networks. In: Cortes, C., Lawrence, N., Lee, D.,

Sugiyama, M., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 28, pp. 2017–2025. Curran Associates, Inc. (2015), https://proceedings.neurips.cc/paper/2015/file/33ceb07bf4eeb3da587e268d663aba1a-Paper.pdf

7. Jai, Y., Hu, S., Martin, R.: Video completion using tracking and fragment merg-ing. The Visual Compute **21**(8-10), 601–610 (2005)
8. Lee, X., Zhang, R., Ebert, F., Abbeel, P., Finn, C., Levine, S.: Stochastic adversarial video prediction.arxivpreprint (2018). https://doi.org/arXiv:1804.01523
9. Lotter, W., Kreiman, G., Cox, D.: Unsupervised learning of visual structure using predictive generative networks.arxivpreprint (2016). https://doi.org/arXiv:1511.06380
10. Pytorchorg: Torch.optim — pytorch 1.6.0 documentation (2020), https://pytorch.org/docs/stable/optim.html, available at:
11. Schüldt, C., Laptev, I., Caputo, D.: Recognizing Human AAction: A Local SVM Ap-proach. ICPR (2004)
12. Shi, J., Tomasi, C.: Good features to track. In: IEEE Conference on Computer Vision and Pattern Recognition (1994)
13. Shi, X., Chen, Z., Wang, H., Yeung, D.: Convolutional lstm network: A machine learning approach for precipitation nowcasting.arxivpreprint. In: Advances in neural information processing systems (2015)
14. Smith, L.: Cyclical learning rates for training neural networks. In: 2017 IEEE Winter Conference on Applications of Computer Vision (WACV) (2015)
15. Vondrick, C., Torralba, A.: Generating the future with adversarial transformers. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR. p. 2992–3000 (2017)

Classifying recognised speech with deep neural networks

Rhyno A. Strydom[0000−0002−4364−2148] and Etienne Barnard[0000−0003−2202−2369]

Multilingual Speech Technologies, North-West University, South Africa;
and CAIR, South Africa.
{rhyno.strydom02, etienne.barnard}@gmail.com

Abstract. We investigate whether word embeddings using deep neural networks can assist in the analysis of text produced by a speech-recognition system. In particular, we develop algorithms to identify which words are incorrectly detected by a speech-recognition system in broadcast news. The multilingual corpus used in this investigation contains speech from the eleven official South African languages, as well as Hindi. Popular word embedding algorithms such as Word2Vec and fastText are investigated and compared with context-specific embedding representations such as Doc2Vec and non-context specific statistical sentence embedding methods such as term frequency-inverse document frequency (TFIDF), which is used as our baseline method. These various embedding methods are then used as fixed length input representations for a logistic regression and feed forward neural network classifier. The output is used as an additional categorical input feature to a CatBoost classifier to determine whether the words were correctly recognised. Other methods are also investigated, including a method that uses the word embedding itself and cosine similarity between specific keywords to identify whether a specific keyword was correctly detected. When relying only on the speech-text data, the best result was obtained using the TFIDF document embeddings as input features to a feed forward neural network. Adding the output from the feed forward neural network as an additional feature to the CatBoost classifier did not enhance the classifier's performance compared to using the non-textual information provided, although adding the output from a weaker classifier was somewhat beneficial.

Keywords: word embeddings · Word2Vec · fastText · Doc2Vec · TFIDF · Deep Neural Networks · CatBoost

1 Introduction

In recent years traditional statistical text representation methods in Natural Language Processing (NLP) are increasingly overshadowed by methods that incorporate Deep Neural Networks (DNNs) to encode information within words, sentences and paragraphs. Where statistical methods have failed at capturing the regularities such as the linguistic structure within languages by representing text as sparse handcrafted feature vectors, DNNs [1] are able to represent text as dense feature vector representations. These representations are able to capture the semantic relationship between words or even different contexts between paragraphs and sentences. Mikolov's "Word2Vec" [2] showed how by representing words as continuous feature representations (word embeddings), it is possible to capture the semantics of words outperforming traditional statistical language modelling techniques such as term-frequency inverse document frequency (TFIDF) [3]. This idea is extended upon with Facebook's "fastText" [4], which learns each character in a word's orthographic representation, helping to generate better word embedding for out-of-corpus words. All these methods, however, are related in that they depend on the same distributional hypothesis which states: "words that appear in the same context share semantic meaning with each other". Mikolov expanded this idea to paragraphs and sentences with "Doc2Vec" [5], representing each document as a dense vector which is trained to predict words in a document.

In this paper, our goal is to investigate these embedding techniques to analyse text produced by a speech recognition system for all the official South African languages and Hindi. Word embeddings are used to determine whether certain keywords of interest in the news have been mentioned falsely or not, due to keywords being mistaken for a similar sounding word in one of the other languages. Identifying these false positives is particularly relevant as it can help advertising agencies know if their advertisements were aired at the right time. This capability can help ensure that these agencies maximise exposure of their clients' products at the most appropriate time of day, and also that media organisations abide by their contractual obligations to broadcast advertisements at agreed upon specified times.

Training word embeddings is an unsupervised learning task, requiring a clearly identified objective in order to evaluate how successful word embeddings are at identifying false positives. Also alterations must be made to the unsupervised learning algorithm based on how well it performs on the task at hand, which is known as extrinsic evaluation [6]. In this paper, we employ the word embeddings as input features to both a linear and non-linear classifier using logistic regression and a DNN classifier, trained to predict whether or not false positive keywords were encountered by the speech recognition system.

We also investigate an alternative method of identifying false positive keywords, where different word embeddings are trained for falsely mentioned keywords (i.e. false positives) and keywords that were correctly identified (i.e. true positives), comparing the cosine similarity between the different occurrences of these keywords.

The main contributions of this work are twofold: on the one hand, we show that text obtained from a speech recognition system provides an interesting and important use case for NLP techniques. On the other, we demonstrate the relative capabilities of various NLP solutions, including embeddings and more traditional methods on such a task in the multilingual South African context.

2 Related work

Text classification is the task of classifying a document to a predefined category. Examples of such task include sentiment analysis, document categorisation and language detection. Linear classifiers such as logistic regression are typically used as a baseline method for NLP problems these classifiers might be very simplistic but can sometimes yield strong baselines for comparing different text classification methods, as shown by Joachims [7] and Fan [8]. Neural networks and support vector machines (SVM) on the other hand are non-linear classifiers and able to distinguish between more complex patterns present in data. Malhar [9] shows how twitter data can be categorised using support vector machines (SVM) outperforming other machine learning methods. Manevitz et al. [10] show how they are able obtain superior results using a simple feed-forward neural network to classify documents on the standard Reuters data-base, comparing it to various traditional machine learning techniques such as Nearest Neighbour, Naive-Bayes and One-Class SVM's. Typically in automatic speech recognition systems (ASR) token identification is used as a means of identifying specific words uttered [11]. Throughout this study we apply more NLP techniques applied to plain text as a mean to recognise text after it has been identified by an ASR system.

Boosting is a method commonly used by machine learning practitioners to reduce both bias and variance of the machine learning algorithm by combining multiple weak learners (i.e. decision tree's) to make one strong classifier. CatBoost [12] is a gradient boosting algorithm developed by Yandex. They show how CatBoost is able to outperform existing state of art boosting algorithms such as XGBoost and LightGBM on various machine learning tasks. We use the output of the neural network as an additional input feature to a CatBoost [12] classifier, comparing how the CatBoost classifier performs with and without the corresponding metadata associated with the text (i.e. keyword or target word, broadcast station name, total duration of the recorded clip in minutes, station language and day of the week). TFIDF is used as a baseline method to evaluate how effectively the embeddings are able to represent the text data

3 Data analysis and exploration

In this section, we investigate the available data and the preprocessing techniques used to clean the text.

3.1 Preprocessing data

Before the text is converted to a numerical representation for the various machine learning algorithms, preprocessing is done on the corpus, shown in fig. 1.
The preprocessing steps are listed as follows:

1. Extract speech-text data from JSON files.
2. Clear the data of any unnecessary symbols and change all the words in the corpus to lower case.

3. Remove English stop-words, i.e. words that do not contribute to the overall meaning in the text, such as 'a', 'the', 'is' and 'are'.
4. Lemmatise the words in text, converting all English words to their base form, for example "geese" becomes "goose" and "caring" becomes "care".
5. Identify common phrases in text such as "New York" or "Net income", instead of having a numerical representation for each word separately, which would effectively increase the vocabulary size.
6. Extract data with different window sizes around individual keywords.

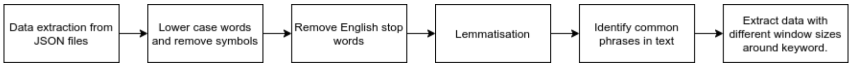

Data extraction from JSON files
Lower case words and remove symbols
Remove English stop words
Lemmatisation
Identify common phrases in text
Extract data with different window sizes around keyword.

Fig. 1: Flow diagram detailing data extraction and preprocessing pipeline.

3.2 General corpus information

The speech-text data was acquired from local South African broadcasting stations and is an amalgamation of radio news and television channel data the data were manually labelled to indicate false positives by people working for the Novus Group.

The text in the data set is predominantly English; there are almost 10 times more English sentences in comparison with the second biggest language in the data set, Afrikaans, as shown in Fig. 2. There are 885,930 speech-text examples (many examples contain more than one keyword) in the corpus of which 675,425 are examples of true positives and 210,505 are true negatives (false positives from the speech-text classifier). This means that 76.24% of the data consists of true positives and 23.76% of true negatives.

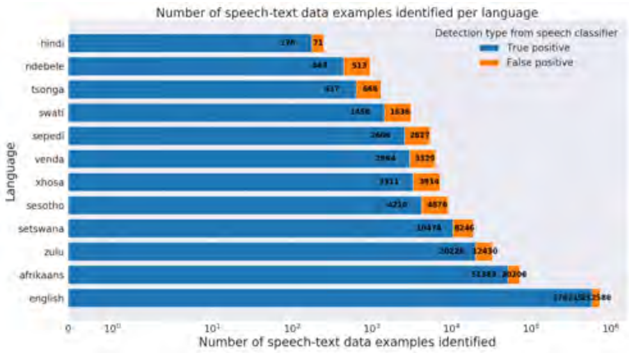


Fig. 2: Frequency of occurrence for different languages in the corpus.

The raw corpus consists of 248,069,944 words with a vocabulary size of 69,052. After pre-processing the word count is significantly reduced to 104,565,491 and the vocabulary size increased to 237,548. There are 4,186 unique keywords, of which 2,729 keywords have examples of where they appear in text as both true and false detection examples. 1,457 examples were only being detected once or more as either true or false.

Keyword pairs are broken into individual keywords with different word window sizes around specific keywords of interest (window size refers to the words to the left and right of the keyword). A new corpus is thus created where each keyword has its own pseudo sentence. This is done to investigate how many context words around specific keywords are needed to achieve good classification. Information regarding this newly generated corpus can be seen in table 1. It is clear from this table that as the window size increases, so does the vocabulary and word count, which is to be expected.

Table 1: Corpus information, showing window sizes, word count and vocabulary size.

Window size around keyword	**Number of words in corpus**	**Vocabulary size**
window 5	7,562,980	**105,685**
window 11	15,521,514	**150,951**
window 21	27,244,705	**184,033**
window 41	46,840,613	**208,273**
window 61	63,174,235	**217,431**
window 121	99,466,572	**226,408**

Fig.3 shows the most frequent keywords found in a corpus with a window size of 121. Splitting the data in this way resulted in some common nouns appearing more frequently, due to the way the data was captured. These common nouns do not contribute much to the identification of specific words and are only a result of how the data was split.

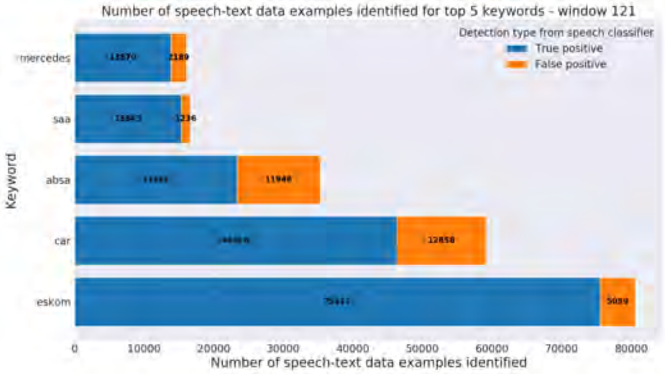


Fig. 3: Top 5 words found in the corpus, with window size of 121.

4 Experimental procedure

In this section, we investigate how word and document embeddings are trained and evaluated with the various classifiers.

4.1 Training, validation and test set

The data is split into a train, validation and test set by randomly splitting the data in accordance to how frequently a specific keyword appears in the corpus — 10% of the data is used for the test and validation set respectively, thus 80% of the data remains for training. We report the accuracy achieved on the test set, and do not attempt to perform statistical significance testing since our goal is to give an indication of the relative performance for various choices of algorithms and parameters, rather than a definite ranking.

4.2 Representing text with word and document vectors

Word2Vec & fastText Word2Vec is used to represent words as continuous vector representations. The algorithm has two variants, namely Continuous Bag of Words (CBOW) and Skip-Gram (SG). CBOW maximises the probability of the target word by looking at the context words, thus given a set of words at its input it aims to predict a specific word at its output. SG follows the inverse rule to this; the technique is designed to predict context given a specific target word. fastText is a technique related to Word2Vec, in that it follows the same distributional hypothesis, and can also be trained using either the CBOW or SG method. fastText claims to be able to generate better word embeddings for out-of-corpus words and morphologically rich languages as it takes into account each character in a word's orthographic representation. This enriches the word embedding, given that orthographic N-grams are able to capture some key structural components of words.

In preliminary investigations the following hyperparameters, shown in Table 2, exhibited good overall performance in extrinsic tasks. These values were empirically selected and correspond with some of the practical values suggested by Miklolov [13]. The window size was changed from 5 to 10 as it showed slightly better results and minimum word count set to 2 to neglect words being removed from South African languages with less detection examples.

Table 2: Word2Vec and fastText hyperparameters.

Parameter	**Value**
Embedding dimension	300
Window size	10
Minimum word count	2
Negative sampling rate	5
Sub-sampling rate	1e-5

Doc2Vec Doc2Vec shows it is possible to represent documents of variable lengths as fixed-length paragraph vectors. These document embeddings can then be used for text classification tasks such as sentiment analysis or predicting falsely detected words given a certain context. Document embedding also come in two variants, namely Distributed Bag of Words of Paragraph Vectors (PV-DBOW) and Distributed Memory Model of Paragraph Vectors (PV-DM). PV-DM averages or concatenates the word and paragraph vectors to predict the next word in context, and the paragraph vector that is learned acts as a memory that remembers what is missing from the current context. PV-DBOW, on the other hand, forces the model to randomly predict sampled words from the paragraph at its output; this method thus ignores word ordering at its output. Both these methods are explored and training done in a semi-supervised fashion, where each document of interest is tagged as either a true or false detection regarding a specific keyword of interest in the news.

In preliminary investigations the following document vector hyperparameters, shown in Table 3, exhibited good overall performance in extrinsic tasks. Hyperparmaters suggested by Lau [14] are used the minimum word count is set to 2 for the same reasons as the word embeddings. The document vectors were trained using a smaller embedding dimension than the trained word vectors, as no improvements were seen in classification increasing this value beyond 100 dimension.

Table 3: Doc2Vec hyperparameters.

Parameter	**Value**
Embedding dimension	100
Window size	15
Minimum word count	2
Negative sampling rate	5
sub-sampling rate	1e-5

Term frequency-inverse document frequency TFIDF is a numerical statistic used to express the importance of a word in a document when considering the whole corpus; the statistic is calculated using equation 1. Term frequency refers to the number of times a specific word appears in a document, while inverse document frequency indicates the importance of the word and is calculated by taking the number of documents in the corpus and dividing it by the number of times a specific word appeared in the corpus. This has the effect of rarer words having a larger numerical value assigned to them, the intuition being that rarer words carry more meaning in a document and should therefore be assigned a larger numerical value than words that appear more frequently and carry no meaning. This method, however, results in a sparse document representation neglecting word order in the sentence. The best results for TFIDF were obtained constraining the vocabulary to 60,000 words for the top N TFIDF scores.

$$w_{i,j} = tf_{i,j} \times \log(\frac{N}{df_i}) \tag{1}$$

where: $w_{i,j}$ = the numerical statistic of a specific word i in document j
$tf_{i,j}$ = term frequency of word i in document j
N = the number of documents
df_i = number of times word i occurs in the document

5 Classifiers, optimization, and metrics used throughout study

5.1 Average cosine similarity method using keyword flags

Inspired by Word2Vec's distributional hypothesis, the average cosine similarity method uses the trained word vectors for classification. The premise is that keywords that are identified as false positives will be surrounded by similarly incorrect words that have nothing to do with the actual keyword. This essentially means that occurrences of words such as "Disney_t" (labelled with a flag "t" for true detection) will be surrounded by completely different words than those of "Disney_f" (labelled with a flag "f" for false detection), as illustrated in fig.4. During inference one would identify the keyword, using a dictionary of all the keywords of interest, and compare the cosine similarity, shown by equation 2, for each word in the sentence for both its true and false keyword examples by adding this flag. The average cosine similarity for both these cases could then be compared with one another, where if the average similarity for "Disney_t" is larger than "Disney_f", the detection is most likely a true detection, and vice versa. The output to this decision is thus binary (therefore a binary classifier). The investigation is done for different window sizes, comparing both fastText and Word2Vec, and investigating both SG and CBOW.

$$\cos\theta = \frac{a \cdot b}{\|a\|\|b\|} \tag{2}$$

where: $\cos\theta$ = cosine similarity between the keyword a and arbitrary word b
a = the keyword of interest for either its true or false examples
b = an arbitrary word in the sentence

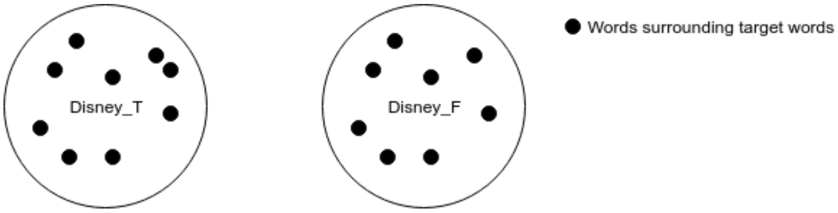


Fig. 4: True detection word embedding cluster for the word Disney (shown left) and false positive word embedding cluster for the word Disney (shown right).

5.2 Logistic regression, DNN and CatBoost classifier

Standard machine learning algorithms require that their input be represented as a fixed length feature vector. Word2Vec and fastText, however, only represent the vectors of individual words. Document vectors are therefore created by taking the average of each word vector in a sentence. Doc2Vec, Word2Vec, fastText and TFIDF are investigated using a logistic regression (linear classifier) and a one layer feed forward neural network classifier (non-linear classifier) with varying widths, also a class weighted logistic regression and feed forward neural network is used, to mitigate the effect of having an unbalanced data set. A heuristic is used to determine the weighting, as shown by equation 3. This weight value is multiplied to each classifier's loss-function, giving more weight to infrequent classes than frequent ones.

After the logistic regression and DNN classifier made its predictions based only on the information provided by text, it is combined with the metadata associated with this prediction, such as the broadcast-station name, keyword of interest, language and time of day, to enhance the classification performance of the CatBoost model, instead of training only on the metadata and ignoring the text associated with it. The same heuristic shown in equation 3 was used to counter the effect of class imbalance.

$$\text{class-weight} = \frac{\text{total number of samples}}{\text{number classes} \times \text{number of samples of specific class}} \tag{3}$$

5.3 Metrics used for evaluation and optimisation

The various classifiers are evaluated based on the Cohen Kappa [15] statistic. It is used to compensate for the effect of the class imbalance present in the data, and it measures the inter-rater agreement for categorical items. To evaluate feature importance in the CatBoost classifier, mean absolute SHAP [16] values are used. The shapely values are calculated by comparing what the model predicts with and without a specific categorical feature, and this is then used to determine which features are relevant and which are not.

Both the CatBoost and DNN classifier use early stopping based on the Area Under the Curve (AUC) value to counter over-fitting. The DNN uses the Adaptive Moment Estimation (Adam) [17] optimiser with a batch size 1024 and a learning rate of 0.001, including a rectified linear unit (ReLU) function before the softmax layer.

6 Results

Using TFIDF, Word2Vec, fastText and Doc2Vec embeddings as input features to a logistic regression (LR) and DNN classifier, the following results were obtained for different word window sizes. It should be noted that initial investigations found that Word2Vec and fastText models trained using the SG method performed better than models trained using the CBOW method. Similarly, Doc2Vec methods trained using the PV-DBOW method performed significantly better compared to PV-DM. The results of these methods are therefore not presented in the following tables, they are, however, present for the average cosine-similarity method.

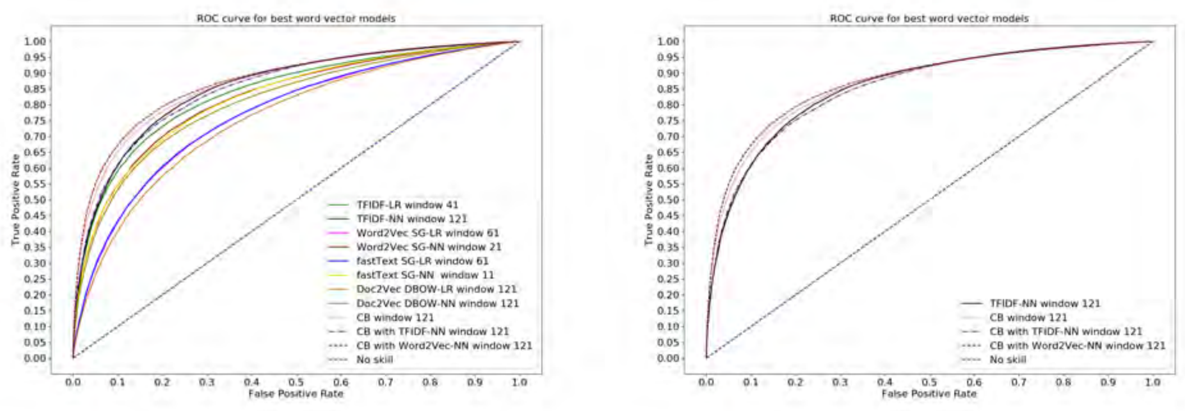
Table 4: Cohen kappa statistic on validation set using optimal window size, for various classifiers .

	Classifiers					
Embedding method	LR	DNN-100	DNN-200	DNN-400	Avg Cosine similarity method	CatBoost
Window size 11						
fastText SG	-	0.38563	0.39127	0.39540	-	-
Window size 21						
fastText SG	-	-	-	-	0.36370	-
Word2Vec SG	-	0.38288	0.39344	0.39876	-	-
Window size 41						
TFIDF	0.43271	-	-	-	-	-
fastText CBOW	-	-	-	-	0.22921	-
Window size 61						
fastText SG	0.32488	-	-	-	-	-
Word2Vec SG	0.32436	-	-	-	0.32971	-
Doc2Vec DBOW	-	0.37808	0.38126	0.38343	-	-
Window size 121						
No embedding (Baseline)	-	-	-	-	-	**0.49639**
TFIDF	-	0.45667	0.47026	0.47310	-	**0.46679**
Word2Vec CBOW	-	-	-	-	0.30704	-
Word2Vec SG	-	-	-	0.36289	-	-
Doc2Vec DBOW	0.30610	-	-	-	-	-

Table 5: Cohen kappa statistic on test set using optimal window size.

	Classifiers					
Embedding method	LR	DNN-100	DNN-200	DNN-400	Avg Cosine similarity method	CatBoost
Window size 121						
No embedding (Baseline)	-	-	-	-	-	**0.49481**
TFIDF	-	-	-	0.47209	-	**0.46465**
Word2Vec SG	-	-	-	0.36128	-	**0.51399**

The Receiver Operating Characteristic (ROC) curves for the logistic regression, DNN and CatBoost classifiers, are shown in fig.5(a)-(b). Note that only the 400 hidden node DNN classifier results are included in the figures.

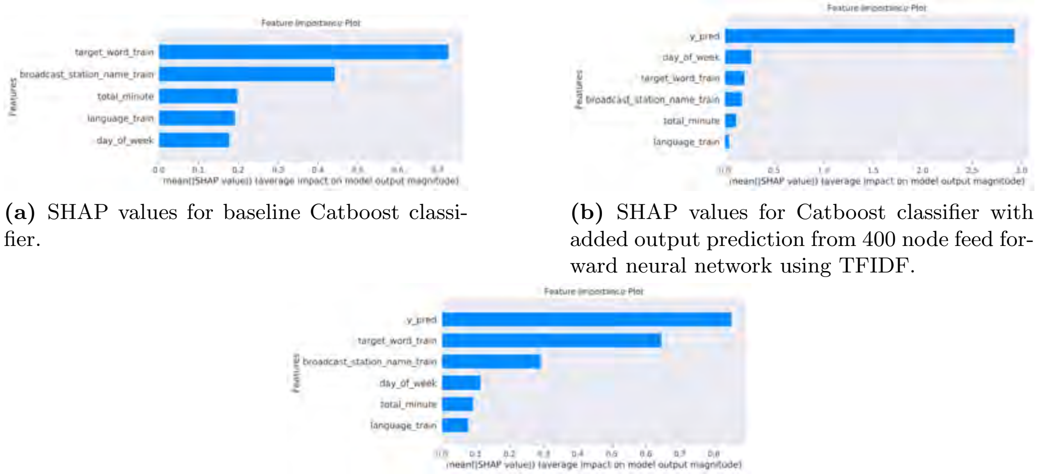


(a) ROC curves evaluated on validation set.

(b) ROC curves evaluated on test set.

Fig. 5: ROC plots for validation train and test set for CatBoost (shown as CB), neural network (shown as NN) and logistic regression (shown as LR) classifier using different embedding techniques.

The mean absolute SHAP values were used to evaluate the contribution of the CatBoost classifier's features to the output. This is shown in fig.6.



(a) SHAP values for baseline Catboost classifier.

(b) SHAP values for Catboost classifier with added output prediction from 400 node feed forward neural network using TFIDF.

(c) SHAP values for Catboost classifier with added output prediction from 400 node feed forward neural network using Word2Vec.

Fig. 6: Mean absolute SHAP values of CatBoost classifier, evaluated on test set.

7 Discussion

When comparing the results of the logistic regression classifier in table 4 with the 100 hidden node feed forward neural network classifier, the logistic regression classifier shows poor performance using Word2Vec, fastText and Doc2Vec as the textual representations in comparison with the neural network. The discrepancy between the two classifiers is due to the neural network's ability to better capture the non-linearity of these embeddings compared to that of the logistic regression classifier. Generally it also seems that larger window sizes for Doc2Vec generate better embeddings for extrinsic tasks, although the same is not always true for Word2Vec and fastText: After a certain point, averaging word vectors over larger window sizes result in some information loss with regards to important words in the context window, which might indicate a false-positive or not. A window size of either 11 and 21 for Word2Vec and fastText regularly produces good results. Increasing the capacity of the feed forward neural network with more hidden nodes, increases the Cohen Kappa metric, although the performance starts to plateau after increasing the neural network size beyond 200 hidden nodes. TFIDF, used as a baseline method to evaluate the word embeddings, outperforms all other embedding methods. This is due to the fact that word and document vectors contain more latent information compared to TFIDF, thus the model might over-fit on features in the textual representations that are less important, whereas TFIDF is able to capture more clearly these important features in text. Training word vectors on a larger text corpus should result in better word embeddings. Furthermore, using larger window sizes with TFIDF (capturing more textual data) tends to improve classification performance.

The baseline CatBoost classifier achieves good results by relying on the target word and broadcast station name as its most important input features, shown using SHAP values in Fig. 6.(a). Adding the output from the 400 hidden node feed neural network classifier using TFIDF method, however, degrades the model's performance, shown in table 5. The model also becomes less reliant on the broadcast station name and target word as important input features, and sees the neural network's output as the most important indicator of whether a false positive or true detection was encountered, shown in Fig. 6.(b), over-fitting the CatBoost classifier to that specific feature. Using the output of the 400 hidden node neural network with Word2Vec (i.e. a weaker model), however, slightly improves the CatBoost classifier performance, with the output from the neural network shown as the most important of the three dominant features (i.e. "y_pred", "target_word_train" and "broadcast_station_name_train"), shown in fig. 6.(c). The broadcast station's language, day of the week and total minutes do not have much effect on the classification result, and these features can thus be removed.

The average cosine similarity method, used to classify words based on true and false flags, achieved better performance than the logistic regression classifier, but worse performance compared to the neural network. Using the CBOW method performance increased as the window size increased, but results were never as good as using the SG method. The optimal window size for this method was 21; increasing it showed no further performance benefit.

8 Conclusion

In this study, different word embedding techniques were investigated employing a DNN and logistic regression classifier, adding the best textual representation output of the DNN classifier as an additional input feature to a CatBoost classifier. It was found that applying TFIDF as the textual representation with a neural network, outperformed all other embedding based methods. TFIDF is able to clearly capture important features in text that the machine learning algorithm can use to discriminate between a false and true positive word. This is due to the limited amount of data and domain specific nature (i.e. broadcast speech-text data) of the task. Training Word2Vec, Doc2Vec and fastText embeddings with a larger corpus results in better word embeddings, increasing the models' chances of detecting a falsely mentioned word. Considering the output of the neural network using TFIDF as an additional feature to the CatBoost classifier degraded its performance. The CatBoost classifier over-fitted on this feature, neglecting additional information provided such as the broadcaster's name and keyword of interest. Considering the output of the DNN using a weaker textual representation method such as Word2Vec slightly enhanced the models performance compared to the baseline CatBoost classifier relying on the metadata associated with the speech-text data. In preliminary investigations using the SG over CBOW based method for Word2Vec and fastText showed slightly better results. This is attributed to SG giving more attention to rarer words in comparison with CBOW which attempts to predict the word given the context. SG furthermore forces the model to learn the context of a specific word. CBOW will thus perform better on a larger training set compared to SG which does not average out peculiarities of a specific word given a context window. The same observations are made using PV-DM with Doc2Vec. It should also be noted that some labelling errors were found, especially with regards to car names, such as "bmw", where the data was labelled as false, but given the context and video clip it is clearly true detection. These types of examples can influence the performance of classifiers trained on such material, reducing the identification accuracy for some keywords of interest.

9 Acknowledgements

We thank the Novus Group for providing the data needed for this investigation. Also a special thanks must be given to Dr. Charl van Heerden at Saigen for facilitating this arrangement and producing the speech-recognition results.

References

[1] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing [Review Article]," *IEEE Computational Intelligence Magazine*, vol. 13, no. 3, pp. 55–75, 2018, ISSN: 15566048. DOI: 10.1109/MCI.2018.2840738.

[2] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *Proceedings of Workshop at ICLR*, vol. 2013, Jan. 2013.

[3] R. Rosenfeld, "Two decades of statistical language modeling: Where do we go from here?" *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1270–1278, 2000.

[4] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching Word Vectors with Subword Information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017. DOI: 10.1162/tacl_a_00051.

[5] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," *31st International Conference on Machine Learning, ICML 2014*, vol. 4, May 2014.

[6] T. Schnabel, I. Labutov, D. Mimno, and T. Joachims, "Evaluation methods for unsupervised word embeddings," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal: Association for Computational Linguistics, Sep. 2015, pp. 298–307. DOI: 10.18653/v1/D15-1036. [Online]. Available: https://www.aclweb.org/anthology/D15-1036.

[7] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *ECML*, 1998.

[8] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, Aug. 2008. DOI: 10.1145/1390681.1442794.

[9] M. Anjaria and R. M. R. Guddeti, "Influence factor based opinion mining of twitter data using supervised learning," *2014 Sixth International Conference on Communication Systems and Networks (COMSNETS)*, pp. 1–8, 2014.

[10] L. Manevitz and M. Yousef, "One-class document classification via neural networks," *Neurocomputing*, vol. 70, pp. 1466–1481, Mar. 2007. DOI: 10.1016/j.neucom.2006.05.013.

[11] D. Ram, A. Asaei, and H. Bourlard, "Sparse subspace modeling for query by example spoken term detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 6, pp. 1130–1143, 2018. DOI: 10.1109/TASLP.2018.2815780.

[12] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "Catboost: Unbiased boosting with categorical features," in *Advances in Neural Information Processing Systems 31*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., Cur-

ran Associates, Inc., 2018, pp. 6638–6648. [Online]. Available: http://papers.nips.cc/paper/7898-catboost-unbiased-boosting-with-categorical-features.pdf.

[13] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds., Curran Associates, Inc., 2013, pp. 3111–3119. [Online]. Available: http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf.

[14] J. H. Lau and T. Baldwin, "An empirical evaluation of doc2vec with practical insights into document embedding generation," in *Proceedings of the 1st Workshop on Representation Learning for NLP*, Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 78–86. DOI: 10.18653/v1/W16-1609. [Online]. Available: https://www.aclweb.org/anthology/W16-1609.

[15] J. Cohen, "A coefficient of agreement for nominal scales," *Educational and Psychological Measurement*, vol. 20, pp. 37–46, 1960.

[16] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., Curran Associates, Inc., 2017, pp. 4765–4774. [Online]. Available: https://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.

[17] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, Dec. 2014.

An Ontology for Supporting Knowledge Discovery and Evolution*

Tezira Wanyana[1,2] [0000−0002−5139−8421], Deshendran Moodley[1,2] [0000−0002−4340−9178], and Thomas Meyer[1,2] [0000−0003−2204−6969]

[1] Department of Computer Science, University of Cape Town, South Africa
[2] Center for Artificial Intelligence Research (CAIR), South Africa.
{twanyana, deshen, tmeyer}@cs.uct.ac.za

Abstract. Knowledge Discovery and Evolution (KDE) is of interest to a broad array of researchers from both Philosophy of Science (PoS) and Artificial Intelligence (AI), in particular, Knowledge Representation and Reasoning (KR), Machine Learning and Data Mining (ML-DM) and the Agent Based Systems (ABS) communities. In PoS, Haig recently proposed a so-called broad theory of scientific method that uses abduction for generating theories to explain phenomena. He refers to this method of scientific inquiry as the Abductive Theory of Method (ATOM). In this paper, we analyse ATOM, align it with KR and ML-DM perspectives and propose an algorithm and an ontology for supporting agent based knowledge discovery and evolution based on ATOM. We illustrate the use of the algorithm and the ontology on a use case application for electricity consumption behaviour in residential households.

Keywords: Intelligent agents · ontology · Knowledge discovery · Abductive theory of method.

1 Introduction

In many real world applications, observational data is continuously being captured from complex and erratic physical and social systems that change over time. For example, in earth sciences, natural processes may vary significantly at different locations and typically change over time [22]. In electricity consumption, the household consumption behavior may change depending on the season (summer or winter), day type (week day or weekend) or changes in the demographic characteristics of a given household [27–29]. An integral part of data analysis and scientific inquiry is the detection of phenomena and the development and evolution of theories to analyse and explain these phenomena [11].

Knowledge Discovery and Evolution (KDE) is of interest to a broad array of researchers in Philosophy of Science (PoS) and Artificial Intelligence (AI), in particular, Knowledge Representation and Reasoning (KR), Machine Learning and Data Mining (ML-DM) and the Agent Based Systems communities. While each

[*] Hasso Plattner Institute for Digital Engineering

of these communities have powerful tools and techniques for different aspects of KDE, they each have different perspectives for acquiring information, representing knowledge, revising and updating knowledge, and synthesizing or combining information. AI techniques can be considered as either top down or bottom up. KR is regarded as a top-down AI approach which uses mathematical modelling tools that adopt logic and probability to acquire, represent, and reason about expert knowledge in some domain. ML-DM are regarded as a bottom-up approach and have well established techniques for analysing vast quantities of data and generating complex classification and prediction models. These communities not only have different research cultures and practices which makes collaboration and interaction difficult but also use terminologies in different ways and to mean different things.

In this paper we explore the use of Haig's recently proposed Abductive Theory of Method (ATOM) [11] as a basis to design a unified conceptual model for KDE. We explore ATOM from both KR and ML-DM perspectives and propose an algorithm and an ontology to drive the cognitive loop of a KDE agent. We demonstrate the application and use of the algorithm and the ontology on a use case application for electricity consumption behaviour in residential households.

The rest of the paper is organised as follows. In Section 2, we describe ATOM and how it aligns with aspects of KR and ML-DM. In this section, we also present an algorithm and a unified conceptual model for KDE. In Section 3, we discuss some related ontologies. Section 4 presents a formalization of knowledge discovery and evolution using an ontology. In Section 5, we demonstrate the application of the proposed ontology to the electricity consumption use case and we discuss, conclude and provide some future directions in Section 6.

2 Knowledge Discovery

2.1 Theories of Scientific Method

Scientific inquiry and knowledge discovery are complex processes. Scientists use a plethora of specific research methods and a number of different investigative strategies when studying their domains of interest [11]. Science is a complex human endeavour which articulates aims that it seeks to realize, applies methods in order to facilitate its investigations and produces facts and theories in its quest to obtain an understanding of the world. The scientific method aims to bring some order to these practices.

There are three major types of inference that are applied in scientific inquiry. These are: deduction, induction and abduction. In deduction, the truth of the premises is a guarantee that the conclusion is true. Induction is based on data; for instance the frequency of an occurrence in the given data. It involves generating universal conclusions from specific data or premises. Abduction appeals to explanatory considerations that do not necessarily follow logically from the premises. In an event that there is evidence E and some candidate explanations $H_1 .... H_n$ for E, H_i is most likely to be true if it explains E better than any of the other explanations [5].

The inductive and hypothetico-deductive theories are commonly regarded as the two main theories of scientific method. In the inductive theory of scientific method, empirical generalizations are discovered in order to create and justify theories at the same time without having to carry out any empirical testing. On the other hand, the hypothetico-deductive method focuses on the researcher acquiring a hypothesis and testing it by checking its predictive success [11]. Some philosophers of science for example Williamson [30] and Mcmullin [21] argue that abduction is the form of inference that is central to the scientific method. Haig presents a so-called broad theory that incorporates a variety of specific research methods in which the prominent type of inference is abduction [11, 12].

2.2 The Abductive Theory of Method (ATOM)

Haig's Abductive Theory of Method (ATOM) systematically assembles strategies and methods for the detection of empirical phenomena and subsequent construction of explanatory theories. ATOM consists of two overarching methods, i.e. phenomenon detection and theory construction.

ATOM starts with the identification, analysis and extraction of patterns from the data. This would typically comprise of the following steps using statistical and analytical tools: initial data analysis, exploratory data analysis, close replication and constructive replication. This process yields phenomena. These are unexplained "relatively stable, recurrent, general features that researchers aim to explain in the data" [11, 12].

Theory construction is used to provide explanations for the phenomena extracted from the data. ATOM applies abduction in the generation and justification of explanatory theories. Theory construction consists of three sub-methods, i.e. theory generation, theory development and theory appraisal. Plausible theories are generated through abductive or explanatory reasoning using methods like exploratory factor analysis, grounded theory and heuristics. The plausible theories are developed through analogical modeling and are appraised by making judgments on the quality of competing explanations which take into account aspects such as simplicity and consistency with other established theories. The process of theory appraisal applies methods like inference to the best explanation and the theory of explanatory coherence.

In ATOM, phenomena are detected from data and phenomena in turn are used to construct theories. Algorithm 1 shows our interpretation of the basic ATOM process. Note that the steps in lines 12-15 can be repeated several times since theories may emerge from a combination of steps 12 and 13.

We settled on ATOM because, unlike the hypothetico-deductive method, where there is no specific approach to theory formulation, ATOM provides a concrete approach for formulating and generating theories. It also aligns well with both top down and bottom up AI techniques. While ATOM emanated from the behavioural sciences it is applicable to a broad array of complex social, physical and socio-technical systems, such as social networking and health information systems.

Algorithm 1: Basic algorithm for the abductive theory of method (ATOM)

```
input: Data $\mathbf{D}$
output: best explanatory theory $\mathbf{t}$

procedure detectPhenomena(D):
    perform initial data analysis on $\mathbf{D}$ to assess data quality
    repeat until phenomena detected
        suggest pattern using exploratory data analysis
        confirm pattern through close replication e.g. cross validation
        generalize pattern through constructive replication
        if stable pattern found
            $\boldsymbol{p} \leftarrow$ generalised pattern;
    end repeat
    return $\boldsymbol{p}$

procedure constructTheory(p):
    generate plausable theories $\boldsymbol{T}$
    develop theories using analogical modeling
    assess and rank competing theories
    $\boldsymbol{t} \leftarrow$ best theory as explanation for p;
    return $\boldsymbol{t}$

main:
    p=detectPhenomena(D)
    t=constructTheory(p)
return $\boldsymbol{t}$
```

2.3 Machine Learning and Data Mining

Machine learning and data mining are two different areas that have been grouped together in this context. This is because they are both data driven and bottom up and they both offer modern techniques for the detection of phenomena from data which is one of the two main processes in ATOM but note that they are not the same. Knowledge discovery from this perspective involves the discovery of new, previously unknown patterns in the data. Independent examples whose characteristics are different from those defined as normal are first characterised as outliers. Robust techniques have been developed for outlier, anomaly and novelty detection [13, 1, 6], where anomalies are viewed as special kinds of outliers in the data which are of interest to the analyst. Anomaly detection seeks the presence of only one example that cannot be explained by the current model while novelty detection [6] seeks the presence of "cohesive and representative examples" not explained in the current model. The detected novelty patterns translate to the phenomena for which explanations are sought in ATOM. Where labeled data is not available clustering techniques can be used for detecting and managing patterns.

For applications involving dynamic systems, the machine learning community investigates algorithms to predict the next state of the system. In the simple case of a univariate time series prediction problem, this could involve building a model trained on historical data to predict the next sequence or trend [20] in the data set. Prediction of the next state of the system, sequence or trend in time series or data streams is important for phenomenon detection. It is useful for determining system change or concept drift[7]. For example, when predictions deviate consistently from unexpected observations, then it is possible that the system has changed and that the model needs to be updated.

2.4 Knowledge Representation and Reasoning

One of the benefits of unifying logic and probability which has been a persistent concern in artificial intelligence and philosophy of science is that logic can be used to specify properties that are required to hold in every possible world and probability provides a way to quantify the weight and ratio of the world that is required to satisfy the property in question [3]. KR as a top down approach in this context refers to the tools and techniques that are applicable in the process of theory construction in order to explain the detected phenomena. These are captured formally during the KDE process by the ontology we propose in this paper. Providing explanatory theories, which is one of the overarching steps of ATOM, requires robust techniques for acquisition, maintenance, revision, update of and reasoning about domain knowledge. KR has the ability to provide support for this using tools that apply techniques such as logic and probability. These are applicable in the generation, development and appraisal of theories in order to select the best explanatory theory.

2.5 A Unified Conceptual Model for KDE

An intelligent agent view brings into perspective the aspect of automatic knowledge discovery and evolution. The agent takes in observations in the form of stimuli from its environment. Its role is to deliberate on the observations it has acquired in order to supply appropriate responses based on its beliefs. It also needs a mechanism to represent and communicate the discovered knowledge in a way that is understandable by other software and human agents. This is required in order to attain reproducibility and unambiguous representation of provenance information. Therefore, there is need to settle on a formal ontology that would be required for representing, communicating and reasoning about aspects of observation induced knowledge discovery.

ATOM aligns with the agent's cognitive loop i.e, stimuli/observations, deliberation and response. Kuhn argues that anomalies are a resource that triggers the knowledge discovery process [18]. ATOM can cater for this since it starts with acquiring empirical phenomena from data objects [11]. When the agent acquires observations from its environment, anomalies are detected and temporarily stored. This is done in order to acquire more anomalous observations since a single anomalous example may not be enough to act as evidence for a phenomenon [6]. This is followed by the process of detecting phenomena from the anomalous observations and theory construction to explain the detected phenomena.

In order to perform KDE activities presented in ATOM, an agent based system (ABS) would have to implement tools and techniques from ML-DM and KR to carry out phenomenon detection and theory construction as illustrated in Fig. 1. Unifying the bottom up and top down perspectives of knowledge discovery into an intelligent agent perspective speaks to our desire for interfacing reactive processing with deliberative processing; one of the things we seek to achieve in our unified perspective. Experiential and reactive processing are assumed to be achieved by data driven probabilistic learning methodologies and deliberative processing is assumed to be handled using reasoning methodologies [3].

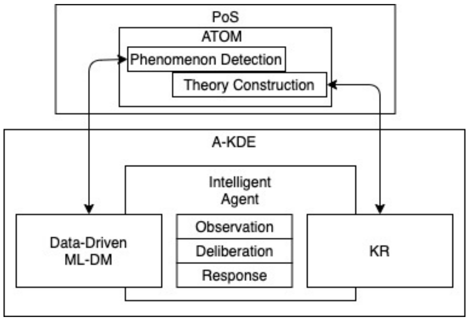


Fig. 1. Knowledge discovery and evolution perspectives

As shown in Fig. 1, ATOM is a unified KDE approach that consists of aspects from both the ML-DM and the KR perspectives. The bottom up (ML-DM) techniques are instrumental in detecting phenomena or novel concepts from the agent's observations and the KR techniques form the foundation of the process of theory construction to explain the detected phenomena in order to fit them into the the body of knowledge that the agent has about the world it inhabits. ATOM forms our stance in this paper in terms of most of the terminology used, evaluation of different ontologies and it also forms the basis of the proposed ontology.

3 Related Ontologies for KDE

In this section, we discuss and evaluate some ontologies for knowledge discovery in terms of the support they offer for observation induced KDE. The evaluation done is in respect to support for capturing aspects of phenomenon detection from data and theory construction to explain the detected phenomena.

Table 1. Comparison of selected ontologies.

ontology has support for:	**LABORS**	**DISK**	**HELO**
Scientific method	Hypothetico - deductive	Hypothetico-deductive	Not explicit
Data	No	Yes, the results produced as a result of executing a workflow from a given line of inquiry	Yes
Phenomena detected from data and phenomenon detection procedure	No	No	Not explicitly stated but records only "Scientific_law" as a statement about phenomena proved by scientific method
Theories that explain phenomena and how they are generated	No	No	Not explicit but records hypothesis and hypotheses set as an explanation and a set of explanations respectively
Models used to elaborate theories	No	No	No
Procedure of theory appraisal	No	No	No
Competing theories	Records research and alternative hypotheses	No	Categorises hypotheses into research, alternatives and negative hypotheses

3.1 LABORS

LABORS (the LABoratory Ontology for Robot Scientists) was designed for representing aspects of scientific experiments for example hypotheses, experimental goals, results, etc. in systems biology and functional genomics. It uses EXPO[3] as an upper level ontology and is used by the robot Scientist [14–17]. The method of discovery that forms the basis of the ontology is hypothetico deductive. The ontology does not include aspects peculiar to knowledge discovery from data or observations and the ontology is very domain specific.

[3] http://expo.sourceforge.net/

3.2 DISK

The DISK (DIscovery of Scientific Knowledge) Ontology [8] for which the requirements are based on the DISK discovery system [9, 10] focuses on representing hypotheses to capture their evolution in automated discovery systems. The hypotheses are supplied by a user or scientist and so there is very little emphasis on the generation of hypotheses or theories. The DISK ontology is constrained to capturing aspects of evolution of user-provided hypotheses.

3.3 HELO

The HELO (HypothEsis and Law Ontology) [26] represents different kinds of scientific statements and links them to their associated probability of being true. It also captures the procedure for obtaining data, research statements and probability statements. The HELO ontology was built from LABORS for the Biomedical domain but it can be used in other domain. Although not explicitly, the HELO ontology provides some support for the KDE ontology requirements as presented in ATOM mainly in the phenomenon detection phase as shown in Table 1. For example, the HELO ontology provides support for *Data*. A concept *scientific_laws* is also used that captures and presents some similarity to *Phenomenon*. The HELO ontology also has a concept similar to *theory* represented as *hypotheses* which captures explanations. However, the nomenclature, taxonomy, interaction and usability of the concepts: *data, phenomenon* and *theory* as represented in the HELO ontology do not capture the aspects and processes in the method that we aim to formalise.

There are other domain specific ontologies designed to support interoperability and reproducibility of scientific investigations and experiments like [24], REPRODUCE-ME [25] for microscopy experiments and OBI (ontology for Biomedical investigations) [2] for biological and medical investigations. Some tools that use ontologies to represent domain knowledge in order to construct theories have also been developed. An example is EIRA (Explaining, Inferring and Reasoning about Anomalies) [23] that was developed for the clinical domain. EIRA [23] does not cater for the representation of hypotheses and their provenance information.

In summary, Table 1 shows a comparison of selected ontologies and the extent to which they offer support for properties of phenomena induced knowledge discovery. The criteria used to evaluate the selected ontologies is extracted from ATOM. The LABORS ontology does not support the criteria used for evaluation in Table 1 because it follows the hypothetico-deductive method specifically in the systems biology and functional genomics domain and focus is on representing aspects of scientific experiments. The same reason applies to the DISK ontology which represents an approach that automates the hypothesise-test-evaluate process that also has characteristics of the hypothetico-deductive method. The HELO ontology, although not explicitly, attempts to capture aspects of the first part of ATOM. However, it does not fully capture the details of the second part. We propose an ontology that caters for the aspects used in Table 1.

4 The KDE Ontology

4.1 Design of the KDE Ontology

The ontology design methodology used was a slight variation of the UPON methodology [4]. UPON is an iterative as well as incremental methodology that is derived from the unified software development process. UPON is use case driven focusing on the development of an ontology that aims to serve either humans or automated systems [4]. The methodology consists of four phases: inception, elaboration, construction and transition phases.

In the *inception phase*, we captured the requirements of the proposed ontology and analysed of existing ontologies. The KDE ontology requirements are mainly based on ATOM [11], a theory of scientific discovery that regards phenomena detected from data as a resource that feeds the knowledge discovery process. The purpose of the ontology is to support an agent based system for knowledge discovery and evolution. The competency questions that the ontology should be able to answer and the household electricity consumption use case were also identified.

A more elaborate analysis was performed in the *elaboration phase* in order to obtain some initial structuring of the main concepts and also to establish the standards to use. We analysed other related ontologies for any reusable concepts and determined how to align the proposed ontology with PROV-O[4], an OWL ontology by W3C provenance working group which provides standards for provenance information.

Design and implementation were the main iterations done in the *construction phase*. Concepts were categorised and the relationships between them established. The ontology was then formalised using OWL (Web ontology language). It incorporates standards provided by PROV [19]. We used Protégé[5], a widely used ontology editing environment to design and implement the ontology.

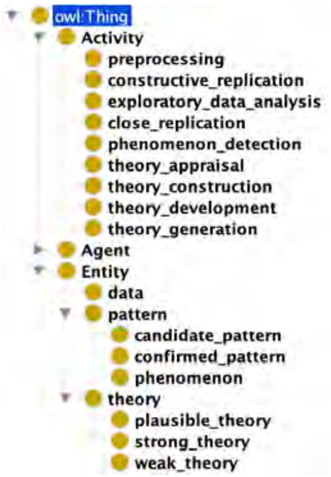
In the *transition phase*, the main activities were around testing the ontology to see if it captures aspects of KDE as presented in ATOM. This involved evaluating the ontology for its support for selected aspects of KDE and checking the ability of the ontology to answer the suggested competency questions.

4.2 The Main KDE Ontology Concepts

In this section , we discuss the main classes represented in the KDE ontology. The ontology is aligned to the W3C PROV standard. The classes, `Entity, Activity` and `Agent` as well as some object properties are drawn from PROV-O. Fig. 2 shows the main classes of the proposed ontology in Protégé. The ontology consists of three main entities and two major activities. The main entities include: `data`, `pattern` and `theory`. The major activities include: `phenomenon_detection` and `theory_construction`. An overview of the main KDE ontology classes and selected object properties is shown in Fig. 3.

[4] https://www.w3.org/TR/prov-o/

[5] https://protege.stanford.edu/



- owl:Thing
  - Activity
    - preprocessing
    - constructive_replication
    - exploratory_data_analysis
    - close_replication
    - phenomenon_detection
    - theory_appraisal
    - theory_construction
    - theory_development
    - theory_generation
  - Agent
  - Entity
    - data
    - pattern
      - candidate_pattern
      - confirmed_pattern
      - phenomenon
    - theory
      - plausible_theory
      - strong_theory
      - weak_theory

Fig. 2. The proposed KDE ontology class hierarchy

Data. Data refers to the instances or information collected for a given purpose. The `data` class captures the required details of the data from which the *pattern* or *phenomenon* is detected. The first activity is to assess the quality of the data using preprocessing (initial data analysis). The activities carried out as part of the preprocessing activity are captured as the class `preprocessing`. The relationship between `preprocessing` and `data` in the ontology is captured using the object property `assesses_quality_of`.

Pattern. The observational evidence required to detect a pattern is provided by data. A pattern is detected from data and this is represented by the object property `was_detected_from` in the KDE ontology. A pattern is an assertion of a recurrent, general feature detected in the data. A phenomenon is a relatively stable pattern, for which an explanation is sought. The phenomenon detection activity captured as `phenomenon_detection` consists of all the tasks undertaken to detect a stable pattern from data. A pattern can be a candidate pattern, a confirmed pattern or a phenomenon captured in the KDE ontology as `candidate_pattern`, `confirmed_pattern` and `phenomenon` respectively. These three types of patterns exhibit a transitive relationship in which a candidate pattern influences a confirmed pattern and in turn, a confirmed pattern influences a stable pattern - the phenomenon. This is captured in our ontology using the `PROV:was_revision_of` object property. A candidate pattern is detected through `exploratory_data_analysis`. This is captured using the object property `was_detected_by`. The candidate pattern is then confirmed through `close_replication`. The object property that captures this relationship is `was_ confirmed_by`. The stability of the confirmed pattern is validated using `constructive_replication`. This is represented using the `was_validated_by`.

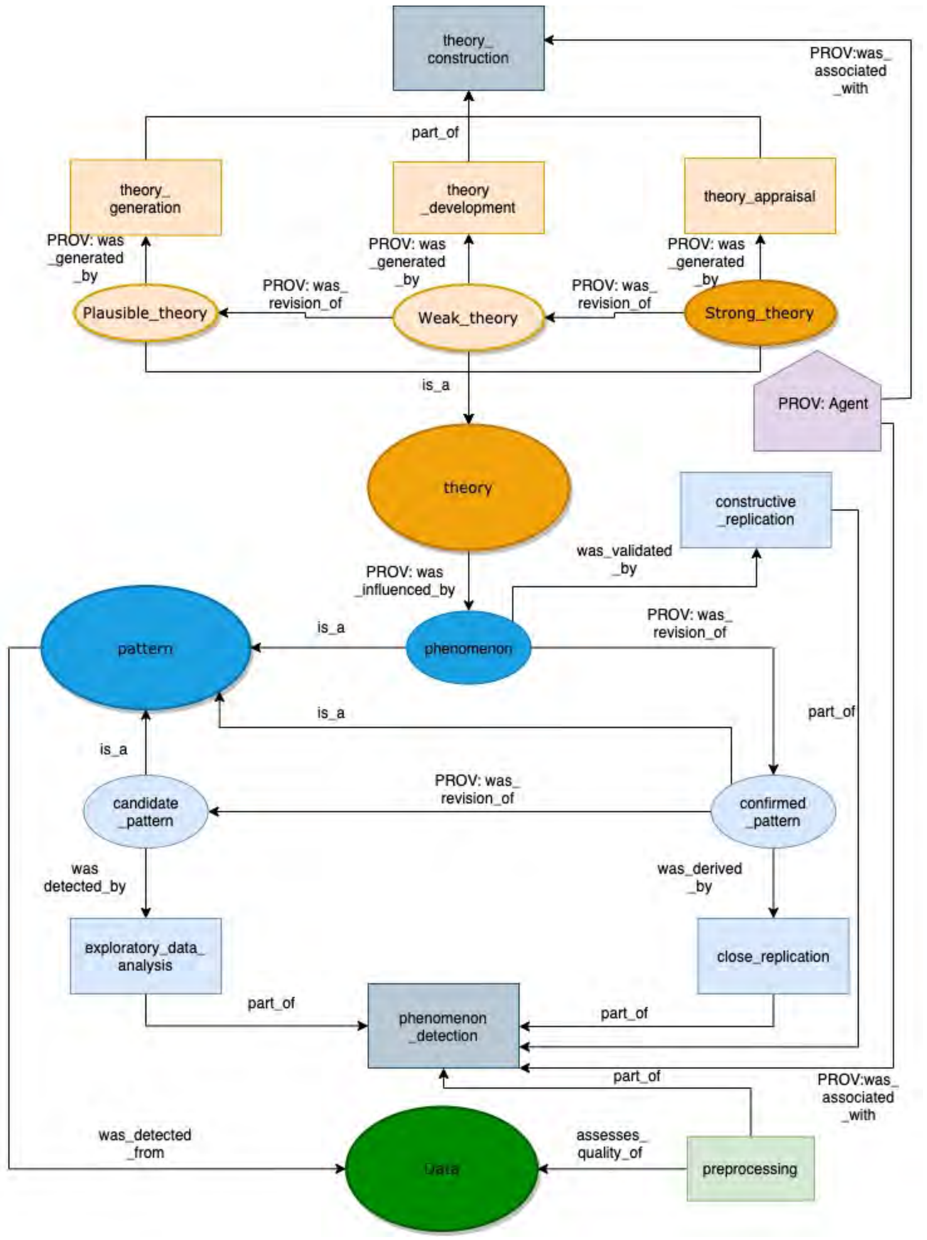


Fig. 3. Overview of the KDE ontology

Theory. This class represents the constructed theories that attempt to explain a given `phenomenon`. The relationship between `theory` and `phenomenon` is represented by the `PROV:was_influenced_by` object property. Theories are only constructed for stable patterns called phenomena. Theories are mainly of three forms; plausible theories, weak theories and strong theories captured as `theory` subclasses: `plausible_theory`, `weak_theory` and `strong_theory`. A weak theory is a revision of a plausible theory and a strong theory is a revision of a weak theory. This relationship is captured in the proposed KDE ontology as `PROV:was_revision_of`. The activity of constructing theories to explain phenomena, captured as `theory_construction` consists of three main subtasks. These include theory generation, theory development and theory appraisal represented as the classes `theory_generation`, `theory_development` and `theory_appraisal`. Theory generation is the process that is used to generate plausible theories. Theory development is used to develop generated theories into weak theories. Theory appraisal represents tasks used to select between competing theories. These aspects as captured using the was the `PROV: was_generated_by`.

4.3 Analysis and Evaluation of the Proposed KDE Ontology

We have presented an ontology that is inspired by the conceptual model for KDE which is based on ATOM. The proposed ontology and the ones briefly discussed in Section 3 formalise the knowledge discovery and evolution process at a meta-level to guide the process of knowledge discovery and evolution. The ontology provides support for the features required for phenomenon detection and theory construction. It also answers the competency questions as required.

The ontology captures features of the data that was used to generate phenomena and the preprocessing that the data was subjected to. Patterns/phenomena detected, at the different stages of stability are recorded along with the techniques used to detect them.

The explanatory theories at each of the levels of theory construction i.e plausible theories, weak theories and strong theories are captured along with the techniques applied during the processes of theory generation, development and appraisal which are all necessary for theory construction.

In conclusion, the proposed KDE ontology captures features, provides support for and distinguishes between data, phenomenon and theory which makes the ontology applicable for agent based KDE from a KR and/ML-DM perspective to record and communicate KDE information. The ontology provides a shared vocabulary and captures the information in a systematic way which aligns with the knowledge discovery process that starts with data or observations, allows for deliberation (phenomenon detection and construction of theories) and return of a response (communicating KDE information). The ontology is accessible online[6].

[6] https://sourceforge.net/projects/akde/

5 Use Case - Household Electricity Consumption

We describe a use case on electricity consumption behaviour of domestic households and use it to illustrate the use of the KDE ontology with two example competency questions answered with SPARQL queries. The use case application is a simplified version of an actual detailed study that used cluster analysis to understand household consumption behaviour in South Africa [27, 28].

Let us assume that the daily electricity consumption of a household h is monitored by collecting hourly data about electricity usage for each day. A 24 element vector is used to represent the consumption daily load profile of the household. Cluster analysis is used to group households with similar daily load profiles for different types of days in the year, depending the season, or whether it is a weekend or weekday. For example there may be different usage on a weekday in summer compared to a week day in winter. The clustering is used to determine the expected consumption behavior of a given household. Each cluster is characterised by the general demographics of the households in it [28, 29]. Demographic data that characterises a cluster could be whether majority of households in the cluster own particular electrical appliances.

Consider two clusters, C_1 and C_2. Let us assume that an agent continuously observes the consumption of household h which it knows to belong to cluster C_1. After some months, through further cluster analysis the agent observes that the daily consumption of household h on summer weekdays increases substantially and now aligns more closely with cluster C_2. This change is identified and captured as a *candidate pattern*. The exploratory method used to generate this pattern and the preprocessing techniques used e.g K-means with unit norm are captured. The agent observes after some time that the increase in consumption for household h persists and it becomes clear that the consumption behaviour now fully aligns with C_2. The *candidate pattern* now becomes and is captured as a *confirmed pattern*. Let us assume that the *confirmed pattern* is then checked for stability by analysing the load consumption of household h again on weekdays during the following year. This *constructive replication* renders the pattern a *phenomenon*. At this point the phenomenon E that the agent seeks to explain is: "h's consumption aligns more closely with cluster C_2 than *C1*"

To provide the best explanatory theory for E, multiple plausible theories $H_1....H_n$ may be generated and developed depending on the agent's beliefs. One of the generated explanations H_i could be: "new appliance ownership". In order to develop H_i further, a comparison is made between the demographic characteristics of C_1 and C_2. One of the differences between C_1 and C_2 is that the majority of households in C_2 own appliances, i.e at least a stove, while the households in C1 do not. The agent may find that with the current information, the most probable explanation is new appliance ownership at household h and the appliance is most probably a stove. Given that this is best explanation from all available explanations, this assertion is added to the knowledge base and mark it as a *weak theory*.

Following the electricity consumption behavior use case, two of the answered competency questions are given below.

CQ1(What theories exists for [phenomenon]?) Assuming the phenomenon in question is *'h's consumption behaviour aligns more closely with C2 than C1'* the following SPARQL query in our ontology returns all the theories that explain the phenomenon.

```
SELECT DISTINCT ?theory
      WHERE { ?theory kdeontology:was_Influenced_By
                  kdeontology:h's consumption aligns more closely with
C2 than C1}
```

CQ2 (From what Data was a given pattern detected?)

This competency question would be answered by the following query :

```
SELECT DISTINCT ?Data
                  WHERE  kdeontology:h's consumption aligns more closely
with C2 than C1} kdeontology:was_detected_from
                  ?Data
```

6 Discussion and Conclusion

In this paper, we have proposed an ontology that focuses on modelling features of KDE based on an algorithm designed from ATOM. The ontology harmonises vocabulary that would be used by an agent based system that applies ML-DM and KR tools and techniques in tandem to a given use case in order to detect phenomena and construct explanatory theories for the phenomena. This would enable the representation and communication generated knowledge.

Knowledge discovery processes that involve both phenomena detection and theory construction benefit from the full breadth of the ontology. However, in some cases, a few concepts of the ontology may be used. An example is in data streams where the goal is to obtain generalizations from anomalies obtained from continuously acquired data. In such a case a novel pattern is detected as a result of multiple occurrence of unexpected instances. The resulting pattern is a confirmed pattern which may be checked for stability by observing it again in a different setting in order to consider it a phenomenon. In this case the aim is to merely look for stable phenomena and not to explain them.

We demonstrated the application of the algorithms and the KDE ontology on an electricity load consumption use case. We showed how the proposed ontology for KDE represents and captures specific features of theory construction for a phenomenon detected in the consumption behaviour of a given household.

For future work, we intend to extend the proposed ontology to include more features peculiar to automatic KDE and to evaluate the ontology's usability and applicability by a KDE agent.

7 Acknowledgements

This work was financially supported by the Hasso Plattner Institute for digital engineering through the HPI Research school at UCT.

References

1. Aggarwal, C.C.: Outlier analysis. In: Data mining. pp. 237–263. Springer (2015)
2. Bandrowski, A., Brinkman, R., Brochhausen, M., Brush, M.H., Bug, B., Chibucos, M.C., Clancy, K., Courtot, M., Derom, D., Dumontier, M., et al.: The ontology for biomedical investigations. PloS one **11**(4), e0154556 (2016)
3. Belle, V.: Logic, probability and action: A situation calculus perspective. In: International Conference on Scalable Uncertainty Management. pp. 52–67. Springer (2020)
4. De Nicola, A., Missikoff, M., Navigli, R.: A proposal for a unified process for ontology building: Upon. In: Andersen, K.V., Debenham, J., Wagner, R. (eds.) International Conference on Database and Expert Systems Applications. pp. 655–664. Springer Berlin Heidelberg (2005)
5. Douven, I.: Abduction. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy. Metaphysics Research Lab, Stanford University, summer 2017 edn. (2017)
6. Faria, E.R., Gonçalves, I.J., de Carvalho, A.C., Gama, J.: Novelty detection in data streams. Artificial Intelligence Review **45**(2), 235–269 (2016)
7. Gama, J.: Knowledge discovery from data streams. CRC Press (2010)
8. Garijo, D., Gil, Y., Ratnakar, V.: The disk hypothesis ontology: Capturing hypothesis evolution for automated discovery. In: K-CAP Workshops. pp. 40–46 (2017)
9. Gil, Y., Garijo, D., Ratnakar, V., Mayani, R., Adusumilli, R., Boyce, H., Mallick, P.: Automated hypothesis testing with large scientific data repositories. In: Proceedings of the Fourth Annual Conference on Advances in Cognitive Systems (ACS). vol. 2, p. 4 (2016)
10. Gil, Y., Garijo, D., Ratnakar, V., Mayani, R., Adusumilli, R., Boyce, H., Srivastava, A., Mallick, P.: Towards continuous scientific data analysis and hypothesis evolution. In: AAAI. pp. 4406–4414 (2017)
11. Haig, B.D.: An abductive theory of scientific method. In: Method Matters in Psychology, pp. 35–64. Springer (2018)
12. Haig, B.D.: The importance of scientific method for psychological science. Psychology, Crime & Law **25**(6), 527–541 (2019)
13. Han, J., Kamber, M., Pei, J.: Data mining: concepts and techniques, waltham, ma. Morgan Kaufman Publishers **10**, 978–1 (2012)
14. King, R.: The adam and eve robot scientists for the automated discovery of scientific knowledge. APS **2017**, X49–001 (2017)
15. King, R.D., Rowland, J., Aubrey, W., Liakata, M., Markham, M., Soldatova, L.N., Whelan, K.E., Clare, A., Young, M., Sparkes, A., et al.: The robot scientist adam. Computer **42**(8), 46–54 (2009)
16. King, R.D., Rowland, J., Oliver, S.G., Young, M., Aubrey, W., Byrne, E., Liakata, M., Markham, M., Pir, P., Soldatova, L.N., et al.: The automation of science. Science **324**(5923), 85–89 (2009)
17. King, R.D., Whelan, K.E., Jones, F.M., Reiser, P.G., Bryant, C.H., Muggleton, S.H., Kell, D.B., Oliver, S.G.: Functional genomic hypothesis generation and experimentation by a robot scientist. Nature **427**(6971), 247–252 (2004)
18. Kuhn, T.S.: The structure of scientific revolutions. University of Chicago press (2012)
19. Lebo, T., Sahoo, S., McGuinness, D., Belhajjame, K., Cheney, J., Corsar, D., Garijo, D., Soiland-Reyes, S., Zednik, S., Zhao, J.: Prov-o: The prov ontology. W3C recommendation **30** (2013)

20. Lin, T., Guo, T., Aberer, K.: Hybrid neural networks for learning the trend in time series. In: Proceedings of the twenty-sixth International Joint Conference on Artificial Intelligence. pp. 2273–2279 (2017)
21. McMullin, E.: The inference that makes science. Zygon® **48**(1), 143–191 (2013)
22. Moodley, D., Simonis, I., Tapamo, J.R.: An architecture for managing knowledge and system dynamism in the worldwide sensor web. International Journal on Semantic Web and Information Systems (IJSWIS) **8**(1), 64–88 (2012)
23. Moss, L., Sleeman, D., Sim, M., Booth, M., Daniel, M., Donaldson, L., Gilhooly, C., Hughes, M., Kinsella, J.: Ontology-driven hypothesis generation to explain anomalous patient responses to treatment. In: Research and Development in Intelligent Systems XXVI, pp. 63–76. Springer (2010)
24. Sahoo, S.S., Valdez, J., Rueschman, M.: Scientific reproducibility in biomedical research: provenance metadata ontology for semantic annotation of study description. In: AMIA Annual Symposium Proceedings. vol. 2016, p. 1070. American Medical Informatics Association (2016)
25. Samuel, S., König-Ries, B.: Reproduce-me: ontology-based data access for reproducibility of microscopy experiments. In: European Semantic Web Conference. pp. 17–20. Springer (2017)
26. Soldatova, L.N., Rzhetsky, A., De Grave, K., King, R.D.: Representation of probabilistic scientific knowledge. In: Journal of Biomedical Semantics. vol. 4, pp. 1–12. BioMed Central (2013)
27. Toussaint, W.: Evaluation of clustering techniques for generating household energy consumption patterns in a developing country. Master's thesis, Faculty of Science, University of Cape Town (2019)
28. Toussaint, W., Moodley, D.: Comparison of clustering techniques for residential load profiles in south africa. In: Davel, M.H., Barnard, E. (eds.) Proceedings of the South African Forum for Artificial Intelligence Research Cape Town, South Africa, 4-6 December, 2019. CEUR Workshop Proceedings, vol. 2540, pp. 117–132. CEUR-WS.org (2019)
29. Toussaint, W., Moodley, D.: Automating cluster analysis to generate customer archetypes for residential energy consumers in south africa. arXiv preprint arXiv:2006.07197 (2020)
30. Williamson, T.: Semantic paradoxes and abductive methodology. In: Armour-Garb, B. (ed.) Reflections on the Liar., pp. 325–346. Oxford University Press Oxford (2017)

Part IV

Knowledge Representation and Reasoning

Knowledge Representation and Reasoning: Abstracts of Full Papers Published in Springer CCIS Volume 1342

The papers in this section appear in Springer CCIS Volume 1342
available at https://link.springer.com/book/10.1007/978-3-030-66151-9.
The abstracts are included in this proceedings.

- Baker, Clayton; Denny, Claire; Freund, Paul and Meyer, Thomas. *Cognitive defeasible reasoning: the extent to which forms of defeasible reasoning correspond with human reasoning.*
- Derks, Iena and de Waal, Alta. *A Taxonomy of Explainable Bayesian Networks.*
- Paterson-Jones, Guy; Meyer, Thomas and Casini, Giovanni. *A Boolean Extension of KLM-Style Conditional Reasoning.*
- Varzinczak, Ivan. *An exercise in a non-classical semantics for reasoning with incompleteness and inconsistencies.*

Cognitive Defeasible Reasoning: the Extent to which Forms of Defeasible Reasoning Correspond with Human Reasoning *

Clayton Kevin Baker[1] [0000−0002−3157−9989], Claire Denny[1] [0000−0002−7999−8699], Paul Freund[1] [0000−0002−2826−6631], and Thomas Meyer[1] [0000−0003−2204−6969]

[1]University of Cape Town and CAIR, South Africa
bkrcla003@myuct.ac.za
dnncla004@myuct.ac.za
frnpau013@myuct.ac.za
tmeyer@cs.uct.ac.za

Abstract. Classical logic forms the basis of knowledge representation and reasoning in AI. In the real world, however, classical logic alone is insufficient to describe the reasoning behaviour of human beings. It lacks the flexibility so characteristically required of reasoning under uncertainty, reasoning under incomplete information and reasoning with new information, as humans must. In response, non-classical extensions to propositional logic have been formulated, to provide non-monotonicity. It has been shown in previous studies that human reasoning exhibits non-monotonicity. This work is the product of merging three independent studies, each one focusing on a different formalism for non-monotonic reasoning: KLM defeasible reasoning, AGM belief revision and KM belief update. We investigate, for each of the postulates propounded to characterise these logic forms, the extent to which they have correspondence with human reasoners. We do this via three respective experiments and present each of the postulates in concrete and abstract form. We discuss related work, our experiment design, testing and evaluation, and report on the results from our experiments. We find evidence to believe that 1 out of 5 KLM defeasible reasoning postulates, 3 out of 8 AGM belief revision postulates and 4 out of 8 KM belief update postulates conform in both the concrete and abstract case. For each experiment, we performed an additional investigation. In the experiments of KLM defeasible reasoning and AGM belief revision, we analyse the explanations given by participants to determine whether the postulates have a normative or descriptive relationship with human reasoning. We find evidence that suggests, overall, KLM defeasible reasoning has a normative relationship with human reasoning while AGM belief revision has a descriptive relationship with human reasoning. In the experiment of KM belief update, we discuss counter-examples to the KM postulates.

Keywords: non-monotonic reasoning · defeasible reasoning · belief revision · belief update · survey · Google Forms · Mechanical Turk.

* Supported by Centre for Artificial Intelligence Research (CAIR)

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_13

A Taxonomy of Explainable Bayesian Networks

Iena Petronella Derks[1][0000−0002−7070−5036] and Alta de Waal[1,2][0000−0001−8121−6249]

[1] Department of Statistics, University of Pretoria
[2] Center for Artificial Intelligence Research (CAIR)

Abstract. Artificial Intelligence (AI), and in particular, the explainability thereof, has gained phenomenal attention over the last few years. Whilst we usually do not question the decision-making process of these systems in situations where only the outcome is of interest, we do however pay close attention when these systems are applied in areas where the decisions directly influence the lives of humans. It is especially noisy and uncertain observations close to the decision boundary which results in predictions which cannot necessarily be explained that may foster mistrust among end-users. This drew attention to AI methods for which the outcomes can be explained. Bayesian networks are probabilistic graphical models that can be used as a tool to manage uncertainty. The probabilistic framework of a Bayesian network allows for explainability in the model, reasoning and evidence. The use of these methods is mostly ad hoc and not as well organised as explainability methods in the wider AI research field. As such, we introduce a taxonomy of explainability in Bayesian networks. We extend the existing categorisation of explainability in the model, reasoning or evidence to include explanation of decisions. The explanations obtained from the explainability methods are illustrated by means of a simple medical diagnostic scenario. The taxonomy introduced in this paper has the potential not only to encourage end-users to efficiently communicate outcomes obtained, but also support their understanding of how and, more importantly, why certain predictions were made.

Keywords: Bayesian network · Reasoning · Explainability.

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_14

A Boolean Extension of KLM-style Conditional Reasoning

Guy Paterson-Jones[1][0000−0001−6935−3910], Giovanni Casini[1,2][0000−0002−4267−4447], and Thomas Meyer[1][0000−0003−2204−6969]

[1] CAIR and Univ. of Cape Town, South Africa
guy.paterson.jones@gmail.com, tmeyer@cair.org.za
[2] ISTI-CNR, Italy
giovanni.casini@isti.cnr.it

Abstract. Propositional KLM-style defeasible reasoning involves extending propositional logic with a new logical connective that can express defeasible (or conditional) implications, with semantics given by ordered structures known as ranked interpretations. KLM-style defeasible entailment is referred to as rational whenever the defeasible entailment relation under consideration generates a set of defeasible implications all satisfying a set of rationality postulates known as the KLM postulates. In a recent paper Booth et al. proposed PTL, a logic that is more expressive than the core KLM logic. They proved an impossibility result, showing that defeasible entailment for PTL fails to satisfy a set of rationality postulates similar in spirit to the KLM postulates. Their interpretation of the impossibility result is that defeasible entailment for PTL need not be unique. In this paper we continue the line of research in which the expressivity of the core KLM logic is extended. We present the logic Boolean KLM (BKLM) in which we allow for disjunctions, conjunctions, and negations, but not nesting, of defeasible implications. Our contribution is twofold. Firstly, we show (perhaps surprisingly) that BKLM is more expressive than PTL. Our proof is based on the fact that BKLM can characterise all single ranked interpretations, whereas PTL cannot. Secondly, given that the PTL impossibility result also applies to BKLM, we adapt the different forms of PTL entailment proposed by Booth et al. to apply to BKLM.

Keywords: Non-Monotonic Reasoning · Defeasible Entailment.

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_15

An exercise in a non-classical semantics for reasoning with incompleteness and inconsistencies

Ivan Varzinczak

CRIL, Université d'Artois & CNRS, France
CAIR, Computer Science Division, Stellenbosch University, South Africa
varzinczak@cril.fr

Abstract. Reasoning in the presence of inconsistencies and in the absence of complete knowledge has long been a major challenge in artificial intelligence. In this paper, we revisit the classical semantics of propositional logic by generalising the notion of world (valuation) so that it allows for propositions to be both true and false, and also for their truth values not to be defined. We do so by adopting neither a many-valued stance nor the philosophical view that there are 'real' contradictions. Moreover, we show that satisfaction of complex sentences can still be defined in a compositional way. Armed with our semantic framework, we define some basic notions of semantic entailment generalising the classical one and analyse their logical properties. We believe our definitions can serve as a springboard to investigate more refined forms of non-classical entailment that can meet a variety of applications in knowledge representation and reasoning.

Keywords: Logic · Knowledge representation · Non-classical semantics.

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_16

Knowledge Representation and Reasoning: Full Papers Accepted for SACAIR 2020 Online Proceedings

The following full papers were accepted for inclusion in this proceedings. These papers can be cited as indicated below adding page numbers and the url to the specific paper.

- Kaliski, Adam and Meyer, Thomas . *Quo Vadis KLM-style Defeasible Reasoning?* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
- Maluleke, Lethabo; Singels, Luca and Biebuyck, Caryn. *A Formal Concept Analysis Driven Ontology for ICS Cyberthreats.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
- Marais, Laurette and Pretorius, Laurette. *Exploiting a Multilingual Semantic Machine Translation Architecture for Knowledge Representation of Patient Information for Covid-19.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.

Quo Vadis KLM-style Defeasible Reasoning?

Adam Kaliski and Thomas Meyer

[1] University of Cape Town
[2] Centre for Artificial Intelligence Research

Abstract. The field of defeasible reasoning has a variety of frameworks, all of which are constructed with the view of codifying the patterns of common-sense reasoning inherent to human reasoning. One of these frameworks was first described by Kraus, Lehmann and Magidor, and is accordingly referred to as the KLM framework. Initially defined in propositional logic, it has since been imported into description and modal logics, and implemented into many defeasible reasoning engines. However, there are many ways in which this framework may be advanced theoretically, and many opportunities for it to be applied. This paper covers some of the most prominent areas of future work and possible applications of this framework, with the intention that anyone who has recently familiarized themselves with this approach may then have an understanding of the kind of work in which they could engage.

Keywords: Artificial Intelligence · Knowledge Representation and Reasoning · Defeasible Reasoning · Nonmonotonic logic

1 Introduction

The KLM framework [40,43] has been the subject of extensions since it was initially defined [12,23,34], as well as been implemented in defeasible reasoning engines [52]. The focus that it has received compared to other defeasible reasoning formalisms [2,48,60] is justified by the KLM framework having three core features: a well defined set of postulates, a preferential semantics, and relative computational efficiency. This enables a large degree of flexibility, as postulates may be dropped, or additional ones enforced.

However, there are both theoretical challenges and potential applications for this form of defeasible reasoning. This paper will attempt to compile some current areas for future work. The various areas and problems covered are not intended to be exhaustive, but is rather a selection of the most prominent fields for future work. The intention of this paper is to provide a brief overview of a selection of problems in this field for those who have a foundational understanding of the field, without necessarily knowing the main candidate areas for novel work.

This paper will first define the base propositional language and concepts in defeasible reasoning, and then describe popular frameworks in the field before describing the focus of this paper: the KLM framework. Chapter 3 will outline some applications for defeasible reasoning and then chapter 4 will outline theoretical work in defeasible reasoning with an eye towards the previously described applications.

2 Background

Although the core principles of this paper are independent of a specific language, in the spirit of the initial definition of the KLM framework the base language of this paper is the propositional logic $\mathcal{L}$, which is formed from a finite set of propositional atoms P, denoted with small Latin letters $p, q, r, ...$, along with the propositional connectives $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$ to form a set of well formed formulas in the usual way, denoted with small Greek letters: α, β, $\gamma ... \in \mathcal{L}$. A knowledge base, $\mathcal{K} \subseteq \mathcal{L}$, is a finite set of well formed formulas. Classical logical consequence, generated by Tarskian semantic entailment, will be denoted with $\vDash$. A consequence operator over some set of statements $\mathcal{K}$, will be denoted $\mathcal{C}n(\mathcal{K}) \subseteq \mathcal{L}$, such that $\mathcal{C}n(.)$ in general is a set of formulas in the language.

Classical entailment, $\vDash$, is informed by the regular semantics for propositional logic. Let $\mathcal{U}$ be the set of all valuations, which are denoted u, v... where each valuation is a function: $\mathcal{L} \mapsto \{T, F\}$ where T and F refer to true and false, respectively. For any formula $\alpha \in \mathcal{L}$, if it is the case that $u(\alpha) = T$, then it is the case that $u \Vdash \alpha$, read "u satisfies α". This can be extended to sets of formulas, such that for some $\mathcal{K}$, then $u \Vdash \mathcal{K}$ if and only if for every $\alpha \in \mathcal{K}$, $u \Vdash \alpha$. Satisfaction then defines classical entailment such that for some $\mathcal{K} \subseteq \mathcal{L}$ and for some $\alpha \in \mathcal{L}$, then $\mathcal{K} \vDash \alpha$ if and only if for every $u \in \mathcal{U}$ such that $u \Vdash \mathcal{K}$, it is the case that $u \Vdash \alpha$.

In classical logic, reasoning is patterned along Tarskian notions of entailment [54,63]. Broadly, Tarski identified three main properties for a reasonable operation for a consequence operator $\mathcal{C}n$:

1. Inclusion: $\mathcal{K} \in \mathcal{C}n(\mathcal{K})$.
2. Idempotence: $\mathcal{C}n(\mathcal{K}) = \mathcal{C}n(\mathcal{C}n(\mathcal{K}))$.
3. Monotonicity: if $\mathcal{K} \subseteq \mathcal{K}'$ then $\mathcal{C}n(\mathcal{K}) \subseteq \mathcal{C}n(\mathcal{K}')$.

Any consequence operator that satisfies the above three properties is referred to as a Tarskian operator. Inclusion simply enforces that everything explicitly stated to be the case is in fact entailed, and idempotence states that the entailment relation should derive all possible inferences given a set of statements. These two properties are relatively uncontroversial, however monotonicity warrants discussion.

Formally, monotonicity states that for any two sets of statements $\mathcal{K}$ and $\mathcal{K}'$ such that $\mathcal{K} \subseteq \mathcal{K}'$, then the set of inferences derivable from $\mathcal{K}$ must also be subsumed by the set of inferences derivable from $\mathcal{K}'$. The intuition is that adding information will never retract a conclusion. If an agent could draw an inference based on their knowledge at some point, then there is no information they could learn that would invalidate that inference. Classical, monotonic reasoning accurately models mathematical reasoning, as there is no defeasibility in mathematics: knowledge is built upon knowledge and there is no provable mathematical theorem that invalidates a previously proven mathematical theorem. However, it is relatively well established that this is not the case with human reasoning [59], as humans frequently revise their beliefs about the world according to new information.

Therefore, monotonic reasoning is necessarily incompatible with defeasibility. Defining a defeasible logic therefore requires dropping the property of monotonicity, and the construction of new properties and axioms so as to accurately model common sense reasoning patterns. These properties and axioms will then inform either a semantics or a proof theory for defining defeasible reasoning.

2.1 Defeasible reasoning

In general, defeasible reasoning is defining reasonable notions of logical consequence about knowledge that corresponds somewhat to the various ways that humans reason on a day to day basis. Handling exceptions in information in a sensible way, while still maintaining an intuitive method of modelling the information in question. The seminal example to illustrate the goal of defeasible reasoning is the following "birds" knowledge base:

1. `Bird` → `Flies`
2. `Penguin` → `Bird`
3. `Penguin` → ¬ `Flies`

Under classical logics and notions of consequence, the above set of formulas entails ¬`Penguin`. If `Penguin` were asserted, then the knowledge base is rendered inconsistent and therefore everything is entailed. Defeasible reasoning formalizes what it means for something to *usually* be the case, and what it means to draw a conclusion from given knowledge that is treated as somewhat speculative, and subject to retraction upon learning more information. If only points 1 and 2 were known in the example above, it would be desirable to draw the inference that penguins fly. However, when point 3 is added, it is then desirable that such an inference is retracted. This mechanism of defeasibility is not present in classical reasoning.

There have been a number of formalisms to capture the patterns of defeasible reasoning. Some of the most popular or well-known formalisms will be briefly covered before describing the framework that is the focus of this paper.

Belief revision Belief revision, first defined by Alchourron, Gärdenfors and Makinson (AGM) [1,2], models an agent's set of beliefs about the world by encoding them as a formal set of statements, referred to as a *belief set*, and defines operations that model the agent adjusting the belief set on receiving new information: revision and contraction. The revision operator models being told that a given statement, perhaps not currently derivable in the belief set, is true, and modifies the belief set such that it incorporates the new statement in a satisfiable way. Contraction is the inverse operation, where a statement is provided with the information that it is not inferred from the knowledge base, and the belief set is modified such that the statement is no longer entailed.

Circumscription Circumscription is a well-known formalism for defeasible reasoning, and one of the first such nonmonotonic logics described. First defined by McCarthy [48] and then revised by McCarthy again [49], it is one of the most expressive defeasible logics. Circumscription defines a predicate over the language that states whether or not a particular individual is normal or abnormal, and furthermore states how, or how not, it is abnormal exactly. This is how circumscription achieves its expressivity: a statement as imprecise as "besides x, there is something else abnormal about y". A drawback is that this places a burden on the modelling process to precisely capture the domain knowledge, including choosing which predicates are atypical, and in what way.

Default logic Default logic represents information as defaults with the intended meaning of "most x's are y's", or "typically a's are b's". It was first described by Reiter [60], and then revised by Reiter and Criscuolo [61]. It was originally devised to enrich first-order logic, by adding the notion that there are default states, assumptions that can be drawn as inferences in the absence of information to the contrary, as a solution to the same core problem of nonmonotonic reasoning: how to most effectively model information containing exceptions. Default logic does so by choosing to address a problem arising from modelling around the exceptions: the problem of inheritance for non-exceptional subclasses. Default logic was defined by Reiter proof-theoretically, but did not have a corresponding semantics. The lack of a model theory means that it can be difficult to choose between different extensions, having to instead rely on intuition about what kind of reasoning is suitable for a given domain [62]. However, Delgrande et al. [30] defined a semantics for default logic, along with a number of extensions.

2.2 KLM approach

The KLM framework was initially defined in propositional logic, and encoded defeasibility in an object-level binary connective, $\mid\!\sim$, that is intended to be read as "is typically" and forms a *defeasible implication*, e.g. the defeasible implication $\alpha \mid\!\sim \beta$ is to be read as "α typically implies β" [40,43]. Therefore, the concept that most birds fly, with some exceptions, can be reasonably represented using the statement $\texttt{bird} \mid\!\sim \texttt{flies}$, which conveys that birds typically fly, and so allows for the possibility that there are birds that do not fly. Contrast to the corresponding statement in classical logics, $\texttt{bird} \rightarrow \texttt{flies}$ which will directly contradict with any flightless birds present in the knowledge base. The inclusion of $\mid\!\sim$ in the language then requires the definition of a new language, $\mathcal{L}_D \coloneqq \mathcal{L} \cup \{\alpha \mid\!\sim \beta | \alpha, \beta \in \mathcal{L}\}$, which is the language created by extending $\mathcal{L}$ with $\mid\!\sim$ such that any two formulas in $\mathcal{L}$ can form a defeasible implication, but $\mid\!\sim$ may not be nested.

A corresponding notion of defeasible inference is also defined and denoted $\mid\!\approx$, and can be read as "defeasibly entails". Any such defeasible entailment relation satisfying the following set of properties referred to as the KLM properties [43,23], presented as follows in $\mathcal{L}_D$, is referred to as LM-rational [23]:

1. (LLE) Left logical equivalence: $\dfrac{\mathcal{K} \mid\!\approx \alpha \leftrightarrow \beta,\ \mathcal{K} \mid\!\approx \alpha \mid\!\sim \gamma}{\mathcal{K} \mid\!\approx \beta \mid\!\sim \gamma}$

2. (RW) Right weakening: $\dfrac{\mathcal{K} \mathrel{|\!\approx} \alpha \rightarrow \beta,\ \mathcal{K} \mathrel{|\!\approx} \gamma \mathrel{|\!\sim} \alpha}{\mathcal{K} \mathrel{|\!\approx} \gamma \mathrel{|\!\sim} \beta}$
3. (Ref) Reflexivity: $\mathcal{K} \mathrel{|\!\approx} \alpha \mathrel{|\!\sim} \alpha$
4. And: $\dfrac{\mathcal{K} \mathrel{|\!\approx} \alpha \mathrel{|\!\sim} \beta,\ \mathcal{K} \mathrel{|\!\approx} \alpha \mathrel{|\!\sim} \gamma}{\mathcal{K} \mathrel{|\!\approx} \alpha \mathrel{|\!\sim} \beta \wedge \gamma}$
5. Or: $\dfrac{\mathcal{K} \mathrel{|\!\approx} \alpha \mathrel{|\!\sim} \gamma,\ \mathcal{K} \mathrel{|\!\approx} \beta \mathrel{|\!\sim} \gamma}{\mathcal{K} \mathrel{|\!\approx} \alpha \vee \beta \mathrel{|\!\sim} \gamma}$
6. (CM) Cautious Monotonicity: $\dfrac{\mathcal{K} \mathrel{|\!\approx} \alpha \mathrel{|\!\sim} \gamma,\ \mathcal{K} \mathrel{|\!\approx} \alpha \mathrel{|\!\sim} \beta}{\mathcal{K} \mathrel{|\!\approx} \alpha \wedge \beta \mathrel{|\!\sim} \gamma}$
7. (RM) Rational Monotonicity: $\dfrac{\mathcal{K} \mathrel{|\!\approx} \alpha \mathrel{|\!\sim} \gamma,\ \mathcal{K} \not\mathrel{|\!\approx} \alpha \mathrel{|\!\sim} \neg \beta}{\mathcal{K} \mathrel{|\!\approx} \alpha \wedge \beta \mathrel{|\!\sim} \gamma}$

Each of the above properties is an encoding of a pattern of reasoning that is reasonable in a defeasible context. *LLE* essentially states that two statements that are classically equivalent should have the same defeasible consequences. *RW* continues what *LLE* started by stating that there is a weak form of transitivity when there is a classical logical dependency: if a statement α is a logical consequence of a defeasible consequence of β, then α is also a defeasible consequence of β. *Reflexivity* is a self-explanatory property, and simply enforces that every statement is a defeasible consequence of itself. *And* and *Or* govern how conjunction and disjunction interacts with defeasible consequence: *And* states that if two different statements are defeasible consequences from the same premises, then the conjunction of the two is a defeasible consequence from those premises. *Or* states that if a statement is a defeasible consequence of two different premises, then it is a defeasible consequence of the disjunction of those premises.

CM and *RM* are defeasible counterparts to classical monotonicity. Monotonicity states that *any* new information will never result in a retraction of an inference. *CM* is a modification of monotonicity for a defeasible language, and states that strengthening the premises α of a defeasible implication with a statement β will never result in the retraction of a defeasible conclusion of α provided that β was one of the defeasible conclusions of α to begin with. This change essentially leaves the door open for novel information, knowledge that was not previously derivable from current facts, to result in an inference being withdrawn. This weakening of monotonicity allows for defeasible statements. However, *CM* is still too weak [43], as it does not allow for certain, intuitive statements to be derived. In the case where the new information has nothing to do with an existing inference, it is possible that the existing inference may not be derived in the presence of the new information, even though there is no reason for it to be withdrawn. This argument motivates the addition of *RM*, which states that any new information that does not conflict with any existing knowledge or inferences will not result in a retraction. This is a stronger property than *CM*, and, in fact, in the presence of *RM*, *CM* is superfluous.

The KLM framework is based on a preferential semantics, where a preferential interpretation, $\mathcal{P} \coloneqq \langle S, \prec, l \rangle$ is defined as a set S of states, a partial order $\prec$ over S, and a mapping $l : S \mapsto \mathcal{U}$ that assigns to every state a valuation in $\mathcal{U}$ [43]. The class of preferential interpretations that correspond to the KLM properties defined above are referred to as *ranked* interpretations, and have the property that the partial order $\prec$ is modular, meaning that $\prec$ forms a total pre-order, and essentially generates a number of "tiers" populated by members of S [43,23]. Therefore, ranked interpretations are often instead characterized in the following way [23]: a ranked interpretation $\mathcal{R} : \mathcal{U} \mapsto \mathcal{N} \cup \{\infty\}$ is a function from the set of valuations $\mathcal{U}$ to the natural numbers with infinity, such that $\mathcal{R}(u) = 0$ for some $u \in \mathcal{U}$, satisfying the convexity property: for every $i \in \mathcal{N}$ such that $\mathcal{R}(u) = i$ for some $u \in \mathcal{U}$, then it is the case that for every $0 \leq j < i$ there is a $v \in \mathcal{U}$ such that $\mathcal{R}(v) = j$. The rank of a valuation in $\mathcal{R}$ essentially encodes how that ranked interpretation credits that valuation. The lower the rank of a valuation, the more normal, or typical, the ranked interpretation views the situation corresponding to the valuation, while valuations with rank ∞ represent impossible situations. A ranked interpretation $\mathcal{R}$ satisfies a defeasible implication $\alpha \mathrel{|\!\sim} \beta$ whenever for every $u \in \mathcal{U}$ such that $u \Vdash \alpha$ and there is no $v \in \mathcal{U}$ such that $v \Vdash \alpha$ and $\mathcal{R}(v) < \mathcal{R}(u)$, then $u \Vdash \beta$. That is, $\mathcal{R} \Vdash \alpha \mathrel{|\!\sim} \beta$ if and only if every valuation that is minimal in $\mathcal{R}$ that satisfies α also satisfies β. This can naturally be extended to knowledge bases: a ranked interpretation $\mathcal{R}$ satisfies a defeasible knowledge base $\mathcal{K}$ if and only if it satisfies every defeasible implication in $\mathcal{K}$. Note that this semantics allows classical formulas to be represented as defeasible implications as well, since for some ranked interpretation $\mathcal{R}$ then: $\mathcal{R} \Vdash \neg\alpha \mathrel{|\!\sim} \bot$ for some $\alpha \in \mathcal{L}$ if and only if $u \Vdash \alpha$ for every $u \in \mathcal{U}$ such that $\mathcal{R}(u) \in \mathcal{N}$. Ranked interpretations are linked to the KLM properties via a representation theorem [43], and every ranked interpretation, $\mathcal{R}$, generates a corresponding defeasible entailment relation $\mathrel{|\!\approx}_{\mathcal{R}}$ that satisfies all the KLM properties such that for some knowledge base $\mathcal{K}$ for which $\mathcal{R} \Vdash \mathcal{K}$ then $\mathcal{K} \mathrel{|\!\approx}_{\mathcal{R}} \alpha \mathrel{|\!\sim} \beta$ if and only $\mathcal{R} \Vdash \alpha \mathrel{|\!\sim} \beta$.

As an example, given the propositional logic over the set of propositions $P = \{p, q, r\}$, then the following is a possible ranked interpretation, $\mathcal{R}$, of some knowledge base $\mathcal{K}$, where valuations are represented as propositions in typewriter text, with a bar over a proposition indicating that it is not satisfied by the valuation:

∞	$\overline{\mathtt{p}}\overline{\mathtt{q}}\overline{\mathtt{r}}$ $\mathtt{p}\overline{\mathtt{q}}\overline{\mathtt{r}}$
1	$\mathtt{pq}\overline{\mathtt{r}}$ $\mathtt{p}\overline{\mathtt{q}}\mathtt{r}$ $\overline{\mathtt{p}}\mathtt{q}\overline{\mathtt{r}}$
0	$\mathtt{pqr}$ $\overline{\mathtt{p}}\overline{\mathtt{q}}\mathtt{r}$ $\overline{\mathtt{p}}\mathtt{qr}$

Then, by way of example, the above ranked interpretation forms a defeasible entailment, $\mathrel{|\!\approx}_{\mathcal{R}}$, such that $\mathcal{K} \mathrel{|\!\approx}_{\mathcal{R}} p \mathrel{|\!\sim} q$ and $\mathcal{K} \mathrel{|\!\approx}_{\mathcal{R}} q \mathrel{|\!\sim} r$.

3 Applications

3.1 Legal reasoning

Legal informatics formalizes laws and regulations such that artificial intelligence and data driven techniques can be used to analyse legal systems [28,37,58]. In particular, much attention has been focused on modelling a set of laws and regulations as a normative system: a set of ordered pairs of the form $\langle condition, consequence\rangle$ [37]. One of the main formalisms for describing a normative system is input/output logics [47], a logic that is built up of ordered pairs as the language. Defeasibility in input/output logics has not been widely studied, however there are other logics used for reasoning about laws that have been enriched with defeasible concepts.

One of the main languages that has been used to reason about laws and regulations is deontic logic [37]. Deontic logic is a type of modal logic where the modal operators, $\Box$ and $\Diamond$, are interpreted as "it is obligatory", or "it is permitted" [64]. This, naturally, is a useful framework for analysing legal problems that are concerned with the distinction between what *is* the case, and what *ought* to be the case. There are two main varieties of deontic logic: standard deontic logic (SDL), and dyadic standard deontic logic (DSDL) [33,58]. Standard deontic logic is made up statements of the form $\Box(\alpha)$, and $\Diamond(\alpha)$ with the intended reading of "α is obligatory" and "α is permitted", respectively. Dyadic standard deontic logic, however, is more expressive by being able to also express statements of the form $\Box(\alpha|\beta)$ and $\Diamond(\alpha|\beta)$ with the reading of "given β then α is obligatory" and "given β then α is permitted", respectively. DSDL is therefore useful for reasoning about scenarios where an agent has acted contrary-to-duty [58], by allowing for reasoning about what ought to be the case even in the case where another norm was violated, which implies a level of defeasibility already baked into DSDL.

The legal domain is a great candidate for defeasible reasoning, as laws and regulations are inherently defeasible. Grossi and Rotolo [37] identify three main areas of defeasibility in the law:

1. Conflicts
2. Exclusionary norms
3. Contributory factors

Conflicts arise where two legal norms both apply and lead to contradictory conclusions. These conflicts can themselves be categorized into three different scenarios: [37]

1. One norm is an exception to the other. This is resolved by *lex specialis* which gives priority to the more specific norm, the exception.
2. There exists a ranking between the norms, for example they could be from different authorities. In this case the conflict can be resolved by *lex superior* which gives priority to the higher ranked norm.
3. The norms could have been enacted at different times. In this case the principle of *lex posterior* will resolve the conflict by giving priority to the norm enacted most recently.

Exclusionary norms are legal norms that provide explicit conditions or methods to make other norms invalid, for example fulfilling criteria to make certain evidence inapplicable.

Contributory factors refer to the set of factors that help decide whether or not a norm is applicable. This is a product of the difficulty of precisely describing what criteria need to be met for some legal issue. For example, determining whether the use of a copyrighted piece of work falls under fair use depends on a number of loosely defined factors [37].

More generally, defeasibility is baked into the legal domain [37]. One possible reason is that legality is driven by human cognition, which is inherently defeasible [59]. Law is also a dialectical exercise where conclusions that may be heavily supported by current norms may be rejected.

Given that the legal sphere inherently contains such defeasibility, it then makes sense to use defeasible reasoning in legal informatics. Deontic logic enriched with defeasibility has been shown to solve well-known paradoxes [28], which suggests that defeasibility can be successfully used to enhance reasoning techniques.

3.2 Programming frameworks

Writing programs based on formal logic has been researched since the 1970s [3,46]. Logic programs are a set of rules that form a theory that corresponds to a knowledge base in a formal language, with the goal of computation rather than theorem proving.

One of the earliest formalisms for logic programming, Datalog was originally a database querying language, but has found far more general applicability [31]. Datalog syntax is reflective of database facts and schemas, and has been successfully applied as a declarative programming language, with use in a variety of fields.

Defeasible datalog [38,55] is an extension to disjunctive datalog [31] in which the the KLM postulates [43] may be expressed. It has been shown [38,55] that defeasible entailment relations may be algorithmically defined, and computed, in defeasible datalog. Specifically, algorithms to compute the rational closure [43], lexicographic closure [42], and relevant closure [22] of a defeasible knowledge base given in an extended defeasible datalog was given by Morris et al. [55], while a general defeasible reasoning system using datalog was defined by Harrison and Meyer [38]. Both of the above were defined syntactically, as algorithms over the statements in the knowledge bases themselves. A useful continuation of this work would be provide semantic characterizations of defeasible entailment relations for datalog, which would provide a platform for comparisons in other formalisms, such as in description logics, and allow for importing work already done into datalog.

Datalog itself has found applicability in artificial intelligence projects, specifically DLV [44] and RDFox [57]. RDFox is a RDF data-store that supports datalog reasoning services. Defeasible datalog implies that defeasible implementations of both RDFox and DLV are possible and worth investigation.

Another extension of datalog is that of datalog ± [20,21], an extension that adds quantification to rule heads while restricting syntax in various ways to improve the complexity. Of particular interest is the application of datalog ± to ontologies, as it is strictly more expressive than the description logic DL-Lite, and the potential for datalog ± to be applied to RDF stores [21]. As a direct result of these, datalog ± has the potential to be applied to the semantic web and other RDF systems such as RDFox [57]. The significance of the above in the context of this paper is the potential to therefore investigate enriching datalog ± with defeasible concepts in the vein of Morris et al. [55] and Harrison and Meyer [38]. Extending defeasible datalog to defeasible datalog ± is a natural theoretical continuation that could yield interesting results.

Another logic programming framework that has close ties to defeasible reasoning is answer set programming (ASP) [9,45]. Built around the concept of answer sets: consistent sets of formulas satisfying the constraints defined by the program, ASP has found many applications, from robotics, to planning, and to bioinformatics [32]. ASP has a fundamental link to defeasible reasoning, as answer sets are essentially consistent extensions to a knowledge base, in much the same way that default logic defines consistent extensions to default theories [9]. ASP programs are, in fact, fragments of default logic [60], and can also be represented as theories in nonmonotonic modal logics [50,51]. These features make ASP a promising formalism in which to translate work done in nonmonotonic logics, such as enriching ASP with object-level defeasibility.

4 Future work

4.1 Syntax Sensitivity

Syntactic entailment relations are defined by directly using the statements in a knowledge base, contrasted with entailment relations generated from a semantics. Syntactic methods introduce the property of *syntax sensitivity.* In short, syntax sensitivity is the property that, given two classically logically equivalent sets of statements that have differing syntax, a defeasible entailment relation will draw a particular inference from only one and not the other [7]. This introduces unpredictability: given the same defeasible entailment relation and logically equivalent, under classical semantics, knowledge bases, then the same inferences would be expected to hold. Therefore, investigating the causes of syntax sensitivity, and ways to avoid it is an interesting area of research.

Baral et al. [5] noted that defeasible reasoning can introduce syntactic sensitivity. The prototypical example of syntax sensitive reasoning is the difference between the two knowledge bases: $\mathcal{K} \coloneqq \{\alpha, \beta\}$, $\mathcal{K}' \coloneqq \{\alpha \wedge \beta\}$. Both $\mathcal{K}$ and $\mathcal{K}'$ are logically equivalent under classical semantics. However, there exists many defeasible entailment relations $\approx$ such that $\mathcal{K} \cup \{\neg\alpha\} \approx \beta$, but $\mathcal{K}' \cup \{\neg\alpha\} \not\approx \beta$ [5,6]. There are possible explanations for this feature to be not undesirable: perhaps α and β are observations from different sources, whereas $\alpha \wedge \beta$ is a single observation. Such a reading of the knowledge base may actually validate syntax sensitivity as a feature, not a bug - the form of the knowledge may be a significant aspect of the reasoning.

Consider the following knowledge bases: $\mathcal{K} \coloneqq \{\texttt{Penguin} \wedge \neg\texttt{Bird} \mid\!\sim \bot, \texttt{Bird} \mid\!\sim \texttt{Flies}, \texttt{Bird} \mid\!\sim \texttt{Wings}, \texttt{Penguin} \mid\!\sim \neg\texttt{Flies}\}$ and $\mathcal{K}' \coloneqq \{\texttt{Penguin}\wedge\neg\texttt{Bird} \mid\!\sim \bot, \texttt{Bird} \mid\!\sim$

$\texttt{Flies} \land \texttt{Wings}, \texttt{Penguin} \mathrel{|\!\sim} \neg\texttt{Flies}\}$. Note that under classical semantics, the *materializations* - defined as $\overrightarrow{\mathcal{K}} \coloneqq \{\alpha \rightarrow \beta \mid \alpha \mathrel{|\!\sim} \beta \in \mathcal{K}\}$, that is the set of classical counterparts for every defeasible implication in a knowledge base - of $\mathcal{K}$ and $\mathcal{K}'$ are logically equivalent, i.e. they are both satisfied by the same set of valuations. Given the lexicographic closure [42] as the defeasible entailment relation, $\approx_{LC}$, then it is the case that $\mathcal{K} \approx_{LC} \texttt{Penguin} \mathrel{|\!\sim} \texttt{Wings}$ and $\mathcal{K}' \not\approx_{LC} \texttt{Penguin} \mathrel{|\!\sim} \texttt{Wings}$. Therefore, the syntax of the knowledge base, while not having any effect under classical semantics, can change what a presumptive defeasible entailment relation will or will not infer.

In the context of KLM-style defeasible reasoning, defeasible entailment relations have a syntactic definition [23]. The more presumptive a defeasible entailment, the more syntactic sensitivity is introduced. This is an issue, as it places a burden on the modelling process to represent the information in such a way as to guarantee the correct inferences. Rather, having consistent behaviour for a particular entailment relation would be more desirable for both implementation and theoretical analysis of an entailment. Such a consistent entailment relation is referred as syntax insensitive, and investigating how to ensure that defeasible syntactic entailments are syntax insensitive is an ongoing, significant area of work.

Syntactic methods are very useful for defining algorithms to perform reasoning tasks, and as such this question has direct consequences on defining defeasible reasoning for logic programming. Specifically, the algorithms presented by Morris et al. [55] and Harrison and Meyer [38] use syntactic methods to define algorithms in datalog for computing defeasible entailment relations. Defining syntax insensitive entailment relations will therefore allow for various presumptive defeasible entailment relations that represent common-sense patterns of reasoning relevant to many domains to be defined in an identical manner, and behave reliably and predictably.

The primary research focuses suggested here for syntax sensitivity would be: is syntax sensitivity a significant property of an entailment relation such that it encodes a meaningful reading of the knowledge base, and is there a corresponding syntax insensitive defeasible entailment for any syntax sensitive defeasible entailment?

4.2 Explanation

Explanation in artificial intelligence is a growing area of interest [41], in part because of the opacity of machine learning techniques. However, it is also well established in logic based techniques [39], where the primary goal is to provide, for each inference, a justification: some minimal subset of the knowledge base from which the inference follows. The goals of explainability in knowledge representation are, broadly speaking, to understand entailments that are not obviously inferred by the knowledge, to fix a possibly bugged, or inconsistent, knowledge base, and to gain some better understanding of a knowledge base with which the user may not have prior experience [39]. So far, the majority of work in explainable AI has been in the context of classical reasoning [41], however there is foundational work on extending explainability to the realm of defeasibility [10,27].

Given a classical propositional knowledge base $\mathcal{K} \coloneqq \{\texttt{Bird} \rightarrow \texttt{Flies}, \texttt{Bird} \rightarrow \texttt{Wings}, \texttt{Penguin} \rightarrow \texttt{Bird}, \texttt{Penguin} \rightarrow \neg\texttt{Flies}, \texttt{Robin} \rightarrow \texttt{Bird}\}$, then any classical reasoning engine will claim that $\mathcal{K} \vDash \texttt{Robin} \rightarrow \texttt{Wings}$. A justification based explanation system will be able to go further and produce a minimal subset of $\mathcal{K}$ satisfying the inference, in this case the set [26]:

- $\texttt{Robin} \rightarrow \texttt{Bird}$
- $\texttt{Bird} \rightarrow \texttt{Wings}$

The above example may be simple, but for knowledge bases on the scale of tens or hundreds of thousands of statements, finding the reasoning behind a given entailment without an explanation engine, can be an excruciating task, if at all feasible [39].

In the context of a defeasible logic, the task of explanation is complicated by the inherent nonmonotonicity: a subset of a defeasible knowledge base may well entail an inference that the whole knowledge base does not. As yet, there is only preliminary work in explanation for defeasible reasoning [10,27], and so therefore it is ripe area for research.

An important aspect to consider for explanation is providing useful justifications in a natural language that considers the intended users [53]. Miller [53] provides a selection of features for a useful, or successful, explanation. Computing justifications for a defeasible inference should take such features into account, as the defeasible nature of an inference can prove both a significant aspect that is worth conveying to the users, while also being challenging to accurately convey. The difference between an inference that is actually classical (and therefore will not be retracted) and an inference that is defeasible, and so is a speculative entailment, may well be important information to deal with in a defeasible explanation engine.

Some key areas of research for defeasible explanation are: implementing and generalizing the work of Chama [26], and comparable work for defeasible logic programming formalisms. Chama [26] provided an algorithm for computing justifications for inferences in the rational closure of a defeasible knowledge base [43], and so a natural follow up would be implementing the algorithm in question in an application. Furthermore, the algorithm in question is designed for justifications of inferences entailed by the rational closure, and so an important project would either be generalizing the algorithm to function for any defeasible entailment, or at least for other specific defeasible entailments such as lexicographic closure [42].

4.3 Expressive logics

The KLM framework was first described using a propositional language [40,43], but there has been much work in implementing KLM-style connectives and semantics to modal logics [17,19] and description logics [11,12,14,15,16,18,24,25,35].

Importing nonmonotonic formalisms into more expressive logics is a natural progression of such work, as defeasibility is a different axis of expressivity on which to enrich description logics and modal logics.

Description logics [4] have a correspondence to the Web Ontology Language (OWL) [18,56] which is used to build various ontologies, such as the Gene Ontology [29], and is also the language that defines the semantic web [8,36]. Therefore, progress made introducing defeasibility in description logics has a direct path to enriching the semantic web with defeasible, common-sense patterns of reasoning, along with ontologies used to compile domain knowledge in general.

While defeasible TBox statements in description logics have been defined with respect to representation and semantics [13,15], an ongoing area of research is that of defeasible ABox reasoning [11,14]. While reasoning with a classical ABox has been defined [11], defining reasoning with respect to a defeasible ABox is an open question [12,35]. Additionally, there is also the opportunity to compile the various ways KLM-style defeasibility has been incorporated in description logics and provide an overview paper.

Investigating defeasible modal logics [16,17,19] has relevance to legal reasoning. As stated in section 3.1, the legal domain contains inherent defeasibility when modelling laws and regulations as they are presented. Since deontic logic is a major language for modelling the legal domain, it seems intuitive that a defeasible deontic logic is worth exploring for its ability to resolve at least some conflicts that arise between factual and normative detachment [28,58]. Some primary research areas for defeasible modal logics include investigating nonmonotonic entailment relations and enriching various specific modal logics with defeasibility [19].

5 Conclusion

This paper is intended to be an overview of the various open sub-fields and research questions regarding the KLM framework for defeasible reasoning. Primary theoretical sub-fields covered are: syntax sensitivity, explanation, and theoretical advancements for more expressive languages such as description and modal logics, with the view towards applications such as legal informatics, and logic programming projects such as RDFox and DLV, and the semantic web.

Work in syntax sensitivity has applications in logic programming projects, as they allow for syntactic formulations of defeasible reasoning to be implemented, with expected behaviours. Explanation has applications in any implementation of defeasible reasoning, by providing justifications for inferences allowing for understanding entailments and repairing defeasible knowledge bases. Defeasible reasoning for description logics has many possible applications, the most obvious being to enrich OWL with defeasibility, which has impact on many projects, including the semantic web. Similarly, defeasible modal logics is an impactful field of work, one application of many would be in legal informatics: enriching deontic logic with defeasibility to resolve well-known paradoxes.

References

1. Alchourrón, C.E., Gärdenfors, P., Makinson, D.: On the logic of theory change: Partial meet contraction and revision functions. The journal of symbolic logic **50**(2), 510–530 (1985)
2. Alchourrón, C.E., Makinson, D.: On the logic of theory change: Contraction functions and their associated revision functions. Theoria **48**(1), 14–37 (1982)
3. Apt, K.R.: Logic programming. Handbook of Theoretical Computer Science, Volume B: Formal Models and Sematics (B) **1990**, 493–574 (1990)
4. Baader, F., Calvanese, D., Mcguinness, D., Nardi, D., Patel-Schneider, P.: The Description Logic Handbook: Theory, Implementation, and Applications (01 2007)
5. Baral, C., Kraus, S., Minker, J., Subrahmanian, V.S.: Combining knowledge bases consisting of first-order theories. Computational intelligence **8**(1), 45–71 (1992)
6. Benferhat, S., Cayrol, C., Dubois, D., Lang, J., Prade, H.: Inconsistency management and prioritized syntax-based entailment. In: IJCAI. vol. 93, pp. 640–645 (1993)
7. Benferhat, S., Dubois, D., Prade, H.: How to infer from inconsisent beliefs without revising? In: IJCAI. vol. 95, pp. 1449–1455. Citeseer (1995)
8. Berners-Lee, T., Hendler, J., Lassila, O.: The semantic web. Scientific american **284**(5), 34–43 (2001)
9. Brewka, G., Eiter, T., Truszczyński, M.: Answer set programming at a glance. Communications of the ACM **54**(12), 92–103 (2011)
10. Brewka, G., Ulbricht, M.: Strong Explanations for Nonmonotonic Reasoning, pp. 135–146. Springer International Publishing, Cham (2019). https://doi.org/10.1007/978-3-030-22102-7_6, https://doi.org/10.1007/978-3-030-22102-7_6
11. Britz, K., Casini, G., Meyer, T., Moodley, K., Sattler, U., Varzinczak, I.: Rational defeasible reasoning for description logics. Tech. rep., University of Cape Town (2018)
12. Britz, K., Casini, G., Meyer, T., Moodley, K., Sattler, U., Varzinczak, I.: Theoretical foundations of defeasible description logics. arXiv preprint arXiv:1904.07559 (2019)
13. Britz, K., Casini, G., Meyer, T., Moodley, K., Varzinczak, I.: Ordered interpretations and entailment for defeasible description logics. Tech. rep., Technical report, CAIR, CSIR Meraka and UKZN, South Africa (2013)
14. Britz, K., Casini, G., Meyer, T., Varzinczak, I.: A klm perspective on defeasible reasoning for description logics. In: Description Logic, Theory Combination, and All That, pp. 147–173. Springer (2019)
15. Britz, K., Meyer, T., Varzinczak, I.: Semantic foundation for preferential description logics. In: Australasian Joint Conference on Artificial Intelligence. pp. 491–500. Springer (2011)
16. Britz, K., Meyer, T., Varzinczak, I.: Normal modal preferential consequence. In: Australasian Joint Conference on Artificial Intelligence. pp. 505–516. Springer (2012)
17. Britz, K., Varzinczak, I.: Defeasible modalities. arXiv preprint arXiv:1310.6409 (2013)
18. Britz, K., Varzinczak, I.: Introducing role defeasibility in description logics. In: European Conference on Logics in Artificial Intelligence. pp. 174–189. Springer (2016)

19. Britz, K., Varzinczak, I.: From klm-style conditionals to defeasible modalities, and back. Journal of Applied Non-Classical Logics **28**(1), 92–121 (2018)
20. Calì, A., Gottlob, G., Lukasiewicz, T.: A general datalog-based framework for tractable query answering over ontologies. Journal of Web Semantics **14**, 57–83 (2012)
21. Cali, A., Gottlob, G., Lukasiewicz, T., Marnette, B., Pieris, A.: Datalog+/-: A family of logical knowledge representation and query languages for new applications. In: 2010 25th Annual IEEE Symposium on Logic in Computer Science. pp. 228–242. IEEE (2010)
22. Casini, G., Meyer, T., Moodley, K., Nortjé, R.: Relevant closure: A new form of defeasible reasoning for description logics. In: Fermé, Eduardoand Leite, J. (ed.) Logics in Artificial Intelligence. pp. 92–106. Springer International Publishing, Cham (2014)
23. Casini, G., Meyer, T., Varzinczak, I.: Taking defeasible entailment beyond rational closure. In: European Conference on Logics in Artificial Intelligence. pp. 182–197. Springer (2019)
24. Casini, G., Straccia, U.: Rational closure for defeasible description logics. In: European Workshop on Logics in Artificial Intelligence. pp. 77–90. Springer (2010)
25. Casini, G., Straccia, U.: Lexicographic closure for defeasible description logics. In: Proc. of Australasian Ontology Workshop. vol. 969, pp. 28–39. Citeseer (2012)
26. Chama, V.: Explanation for defeasible entailment. Master's thesis, Faculty of Science (2020)
27. Chama, V., Meyer, T.: Explanation for defeasible entailment (2019)
28. Chingoma, J., Meyer, T.: Forrester's paradox using typicality (2019)
29. Consortium, G.O.: The gene ontology resource: 20 years and still going strong. Nucleic acids research **47**(D1), D330–D338 (2019)
30. Delgrande, J.P., Schaub, T., Jackson], W.K.: Alternative approaches to default logic. Artificial Intelligence **70**(1), 167 – 237 (1994). https://doi.org/https://doi.org/10.1016/0004-3702(94)90106-6, `http://www.sciencedirect.com/science/article/pii/0004370294901066`
31. Eiter, T., Gottlob, G., Mannila, H.: Disjunctive datalog. ACM Transactions on Database Systems (TODS) **22**(3), 364–418 (1997)
32. Erdem, E., Gelfond, M., Leone, N.: Applications of answer set programming. AI Magazine **37**(3), 53–68 (2016)
33. Gabbay, D., Horty, J., Parent, X., van der Meyden, Ron van der Torre, L.: Handbook of Deontic Logic and Normative Systems (01 2013)
34. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.L.: Semantic characterization of rational closure: From propositional logic to description logics. Artificial Intelligence **226**, 1–33 (2015)
35. Giordano, L., Gliozzi, V., Olivetti, N., Pozzato, G.L., et al.: Preferential vs rational description logics: which one for reasoning about typicality?. In: ECAI. pp. 1069–1070 (2010)
36. Grau, B.C., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P., Sattler, U.: Owl 2: The next step for owl. Journal of Web Semantics **6**(4), 309–322 (2008)
37. Grossi, D., Rotolo, A., et al.: Logic in the law: A concise overview. Logic and Philosophy Today. College Publications (2011)
38. Harrision, M., Meyer, T.: Rational preferential reasoning for datalog (2019)
39. Horridge, M.: Justification based explanation in ontologies. The University of Manchester (United Kingdom) (2011)
40. Kraus, S., Lehmann, D., Magidor, M.: Nonmonotonic reasoning, preferential models and cumulative logics. Artificial intelligence **44**(1-2), 167–207 (1990)

41. Krötzsch, M., Stepanova, D.: Reasoning Web. Explainable Artificial Intelligence: 15th International Summer School 2019, Bolzano, Italy, September 20–24, 2019, Tutorial Lectures, vol. 11810. Springer Nature (2019)
42. Lehmann, D.: Another perspective on default reasoning. Annals of Mathematics and Artificial Intelligence **15**(1), 61–82 (1995)
43. Lehmann, D., Magidor, M.: What does a conditional knowledge base entail? Artificial intelligence **55**(1), 1–60 (1992)
44. Leone, N., Pfeifer, G., Faber, W., Eiter, T., Gottlob, G., Perri, S., Scarcello, F.: The dlv system for knowledge representation and reasoning. ACM Trans. Comput. Logic **7**(3), 499–562 (Jul 2006). https://doi.org/10.1145/1149114.1149117, `https://doi.org/10.1145/1149114.1149117`
45. Lifschitz, V.: Answer set programming. Springer (2019)
46. Lloyd, J.W.: Foundations of logic programming. Springer Science & Business Media (2012)
47. Makinson, D., Van Der Torre, L.: Input/output logics. Journal of philosophical logic **29**(4), 383–408 (2000)
48. McCarthy, J.: Circumscription—a form of non-monotonic reasoning. Artificial intelligence **13**(1-2), 27–39 (1980)
49. McCarthy, J.: Applications of circumscription to formalizing common sense knowledge. Artificial Intelligence **28**, 89–116 (1986)
50. McDermott, D., Doyle, J.: Nonmonotonic logic 1. Artificial Intelligence **13**, 41–72 (1980)
51. McDermott, D.: Nonmonotonic logic ii: Nonmonotonic modal theories. Journal of the ACM (JACM) **29**(1), 33–57 (1982)
52. Meyer, T., Moodley, K., Sattler, U.: Dip: A defeasible-inference platform for owl ontologies. CEUR Workshop Proceedings (2014)
53. Miller, T.: Explanation in artificial intelligence: Insights from the social sciences. CoRR **abs/1706.07269** (2017), `http://arxiv.org/abs/1706.07269`
54. Monk, J.D.: The contributions of alfred tarski to algebraic logic. The Journal of Symbolic Logic **51**(4), 899–906 (1986), `http://www.jstor.org/stable/2273903`
55. Morris, M., Ross, T., Meyer, T.: Defeasible disjunctive datalog. pp. 208–219. CEUR (03/12-06/12 2019), `http://ceur-ws.org/Vol-2540/FAIR2019_paper_38.pdf`
56. Motik, B., Patel-Schneider, P.F., Parsia, B., Bock, C., Fokoue, A., Haase, P., Hoekstra, R., Horrocks, I., Ruttenberg, A., Sattler, U., et al.: Owl 2 web ontology language: Structural specification and functional-style syntax. W3C recommendation **27**(65), 159 (2009)
57. Nenov, Y., Piro, R., Motik, B., Horrocks, I., Wu, Z., Banerjee, J.: Rdfox: A highly-scalable rdf store. In: Arenas, M., Corcho, O., Simperl, E., Strohmaier, M., d'Aquin, M., Srinivas, K., Groth, P., Dumontier, M., Heflin, J., Thirunarayan, K., Staab, S. (eds.) The Semantic Web - ISWC 2015. pp. 3–20. Springer International Publishing, Cham (2015)
58. Parent, X., van der Torre, L.: Detachment in normative systems: Examples, inference patterns, properties. IfCoLog Journal of Logics and their Applications **4**(9), 2295–3039 (2017)
59. Pollock, J.L.: Cognitive carpentry: A blueprint for how to build a person. Mit Press (1995)
60. Reiter, R.: A logic for default reasoning. Artificial intelligence **13**(1-2), 81–132 (1980)
61. Reiter, R., Criscuolo, G.: Some representational issues in default reasoning. Computers & Mathematics with Applications **9**(1), 15–27 (1983)

62. Shoham, Y.: A Semantical Approach to Nonmonotonic Logics, p. 227–250. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1987)
63. Tarski, A.: On the concept of logical consequence. Logic, semantics, metamathematics **2**, 1–11 (1936)
64. Von Wright, G.H.: Deontic logic. Mind **60**(237), 1–15 (1951)

A Formal Concept Analysis Driven Ontology for ICS Cyberthreats

Luca Singels[0000−0001−6845−543X], Caryn Biebuyck[0000−0002−3846−6952], and Lethabo Maluleke[0000−0002−0549−5394]

Stellenbosch University, Stellenbosch 7600, South Africa

Abstract. Industrial Control Systems (ICS) are increasingly vulnerable to cyber attacks and there is little in the way of research regarding automated cyber protection capabilities. Significant strides have been made recently by way of expanding the ICS body of knowledge through the release of the MITRE ATT&CK framework for Industrial Control Systems. This paper focuses on creating a meaningful representation of threat techniques and procedures in order to enable practitioners a means for earlier and strategic detection of ICS cyberthreats. This is achieved by applying formal concept analysis to an existing taxonomic framework in order to retrieve deeper conceptualisations and inter-relationships. Thereafter, the results of the exploratory analysis will be encoded to a formal ontology using the OWL ontology language. By applying reasoning and enriching the semantics of cyber techniques and procedures, as recognised by MITRE, a better understanding of the associations and implications of adversary techniques and procedures can be achieved in order to offer practitioners an understanding of what types of attacks are being performed in a system, how and what the next adversary action will likely be. The outcome of this paper is a validated formal ontology capable of being implemented in a system designed for the protection of ICS. Moreover, this paper serves to expand the relatively underdeveloped body of knowledge in the field of ICS and automated cyber intelligence.

Keywords: Formal Concept Analysis · Industrial Control Systems · Cyber-security.

1 Introduction

As industrial control networks become increasingly dynamic and reliant on computerised information systems, exposure to cyberthreats and security risks is intensified. This is particularly worrying as modern society is largely dependent on specialised networks of interdependent IT systems and services, collectively referred to as ICS, for vital functions [29]. Given the importance of these entities, their manipulation or destruction could have detrimental consequences for public well-being, safety and economic security.

Consequently, it is important that service-providers understand the landscape of the applicable cyberthreats so that appropriate security protocols can be implemented to protect the systems controlling these vital human resources. This awareness is stifled as a result of ICS security threats being a relatively underdeveloped domain. Due to this knowledge gap, an urgent need for more in-depth mechanisms of protecting ICS is required. Thus, effective information sharing in regard to ICS cyberthreats is needed in order to help practitioners and researchers identify adversary actions within their systems as well as likely future adversary actions and in doing so create the basis for reducing the risk of ICS being compromised.

Research to date reflects a trend of rising information and knowledge exchange between reliable stakeholders in order to assist the management of system vulnerabilities and incidents and to alleviate threats [8, 9, 30]. By perpetuating a culture of information sharing among the owners and operators of ICS infrastructures, cybersecurity can be improved within and across institutions as insights and improved practices can be gained from the reports of cyber-intrusions of similar systems which may allow organisations to contain security breaches or prevent similar attacks [29]. Accordingly, effective information sharing can enable ICS operators to improve detection time and deploy strategic responses to cyberthreats.

For effective information sharing, the development of standardised terms, taxonomies and distribution mechanisms are required. As a result, practitioners often turn to the development of ontologies as they facilitate effectual knowledge representation and exchange [24, 25]. An ontology, according to [13], is “an explicit specification of conceptualisation,” meaning an ontology elucidates the concepts within a domain of interest as well as the relations between those concepts in a formal language [4, 25]. Obitko et al. [25] are of the view that enhanced knowledge representation and knowledge exchange can occur as ontologies allow concepts to be explained by the relationships applicable to the concept as well as by its location in a taxonomic hierarchy. Accordingly, this paper will use an ontological approach to investigate and represent the ICS cyberthreat environment.

Literature today, concerning knowledge discovery and processing, posits several approaches for designing ontologies. A well-known approach is that of Formal Concept Analysis (FCA). FCA is a philosophy for data analysis that classifies conceptual structures among datasets [25]. In doing so, entities of interest are portrayed in a manner that is suitable for information retrieval, processing, representation and subsequently exchange [25].

By utilising an FCA approach, the intention of this paper is to simplify the identification of ICS cyberthreats through a better understanding of the associations and implications of adversary techniques and procedures. Thereby, we hope to offer practitioners an understanding of what types of attacks are being performed and how, what the next threat will be and as a result allow for significant decreases in detection time and faster mitigated responses.

The remainder of this paper is structured as follows: an extensive literature review is presented to provide background information on the domains of interest, namely, ICS, MITRE’s ATT&CK framework, FCA and ontology engineering and the application thereof. The emerging problem statement is then outlined along with the research objectives of the paper. Thereafter, the methodology utilised for data exploration is clearly presented, and is followed by the reporting of the data analysis. Finally, the resulting ontology is presented together with proposals of potential future research domains.

2 Background and Related Works

2.1 Industrial Control Systems(ICS)

ICS encompass several types of control networks such as supervisory control and data acquisition (SCADA), distributed control systems (DCS) and programmable logic controllers (PLC), which are primarily found in industrial sectors and critical infrastructures [31]. Their core purpose is to monitor and control processes in a variety of industries such as in power plants, oil and gas [18].

Historically ICS were proprietary enclosed systems meaning they were generally isolated with reduced exposure to cyber security threats [31]. However, as the use of complex distributed network architectures increased and with the wider adoption of internet and communication protocols, ICS have faced amplified security risks. It is also important to note that vulnerabilities are found at different levels of the infrastructure – namely the hardware layer, firmware layer, software layer and the network layer – with the hardware layer vulnerabilities affecting the entire life cycle of an ICS from the design to decommission phase. The vulnerabilities associated with the hardware layer are brought about by security concerns in processor supply chains where malware can be potentially injected into the processors [18].

Identified root causes of ICS security vulnerabilities include: (1) poorly secured legacy systems, (2) remote access (3) lack of trained specialists and (4) lack of cybersecurity situational awareness [11]. Adversaries take advantage of these vulnerabilities and configuration flaws to bypass existing security programmes in order to penetrate and compromise ICS. Motive for gaining admission to ICS vary, with access to proprietary data, destruction and theft either for financial or tactical gain being most notable.

To meaningfully understand how ICS networks are vulnerable to adversary action, it is of value to understand their logical architecture, which has been base lined in the Purdue Model Control Hierarchy framework [32]. By displaying the interconnections and inter dependencies between the core components of a classic ICS, the Purdue Model has become a standard reference model for industry stakeholders [2]. The Purdue Model recognises six levels of operation within four zones, although, an extensive overview of each operational level is beyond the scope of this paper. A synopsis of each level, including example devices and utilities is detailed by Ackerman [2].

2.2 MITRE's ATT&CK Framework

The MITRE ATT&CK framework is a global knowledge base curated from real world studies of adversary tactics and techniques that is used as a basis for creating models of cyberthreats. These models can be used to both perform and protect against cyberthreats on wide variety of systems like Mobile, Enterprise and of particular importance to this paper, the relatively new, ICS [32]. The aim of ATT&CK, as specified by Strom et al. [32], is to classify adversary behaviour in order to help improve post-compromise detection of advanced persistent threats.

Alternatives to ATT&CK Besides ATT&CK, there are various other cybersecurity knowledge bases available today. Lockheed-Martin Cyber Kill Chain (LMCKC)[22] and Microsoft STRIDE [1] are some popular examples of alternative threat models. Overall, they provide a very high-level understanding of cyberthreat processes and objectives. The MITRE ATT&CK framework differs from this in that it provides more abstract and in-depth insights into the individual adversary actions, relations between these actions and sequence of actions. This is key in differentiating it from other threat models [32].

Contrary to the high-level tools, there exist databases of malware and exploits, but these are more complex to work with and distant to actual scenarios in which they can be used [32]. A common example of one of these tools is the Vocabulary for Event Recording and Incident Sharing [34]. It is for the aforementioned reasons that the MITRE ATT&CK framework was chosen for further study.

Detailing of the ATT&CK Framework for ICS Up until recently the MITRE ATT&CK framework only operated within the top two levels of the Purdue model with the ATT&CK Framework for Enterprise. The inception of the MITRE ATT&CK framework for ICS, however, introduced inspection of the bottom three levels, with these levels being the area of interest in this paper.

MITRE's ATT&CK framework for ICS acknowledges eleven tactics which condense into the following execution goals, namely, Initial access, Collection, Command and control, Discovery, Evasion, Execution, Impact, Impair process control, Inhibit response function, Lateral movement and Persistence [20]. Moreover, 81 adversary techniques fall within these categories, each identifiable using a unique four-digit identification number starting with 'T8'. In addition to these general forms of attack, procedures which detail the specific software used to execute these attacks are disguised in the knowledge base, and possess IDs starting with 'S0'. Information pertaining to known sets of related invasion activity or 'groups' are also traceable using the ICS knowledge base. Furthermore, specification of the asset categories affected are also addressed within the knowledge base, allowing practitioners to quickly understand whether the listed techniques are pertinent to their specific environment. Although comprehensive, it is important to note that unlike the frameworks pertaining enterprise and mobile, the ATT&CK framework for ICS, as of yet, does not provide a navigable knowledge-base which limits usefulness for practitioners.

2.3 Formal Concept Analysis

FCA Application in Knowledge Discovery and Representation Literature today [36, 33, 5] acknowledges that FCA allows for visualisation which eases the analysis and interpretation of data, which is fundamental for data exploration and subsequently knowledge discovery and representation [28]. By using formal concept lattices to organise information and implicit data connections, FCA provides a clear structure for the transformation of data [36] and for knowledge representation [33,16,17].

FCA Terms and Notions Poelmans et al. [28] explain that the core principle of FCA is that of a formal context which encompasses a triple of sets, $k = (G, M, I)$, with binary relation $I \subseteq G \times M$. This relationship can be embodied in a cross table consisting of a set of rows G (representing objects) and a set of columns M (representing attributes), with crosses representing incidence relation I. Given a formal context, one can derive formal concepts, which can be ordered in a subconcept-superconcept relation. The notion of a formal concept is comprised of an extent and its intent [28]. The extent comprises the set of objects included in the concept whereas the intent comprises the set of attributes.

The set of all concepts ordered by the subconcept-superconcept relation makes a complete lattice, that can be visualised using a line diagram. Line diagrams offer an opportunity to retrieve the extent of a formal concept by tracing all objects on all down-leading paths from the respective node representing the formal concept of interest [28]. Similarly, the intent of a formal concept can be recognised by following all upward-leading paths of the respective node to collect all attributes. Further insights regarding FCA exploration can be gained from the materials developed by Ignatov, Ganter and Obiedkov [14,10].

Association Rules In order to analyse concept lattices, a basic understanding of association rules is required. The relationships between the different formal concepts within a formal context can be described by different levels of support and confidence, measured as values between 0 and 1. The support(*supp*) of a relationship represents the frequency of the relationship within all concept transactions of the formal context, meaning the probability that a transaction contains both concepts of interest [27]. Confidence of association (*conf*) on the other hand, measures the probability of transactions containing a concept also containing another object of interest. The following definitions are vital for understanding concept lattices as well as the relationships between concepts and their attributes and objects [14]:

Definition I. *The relative* ***support*** *of an association rule $A \to B$, denoted supp($A \to B$) shows which part of the set G contains $A \cup B$.The rule is defined as follows:*

$$supp(A \to B) = \frac{\mid (A \cup B)' \mid}{\mid G \mid} \tag{1}$$

Definition II. *The relative* ***confidence*** *of an association rule $A \to B$, denoted, conf($A \to B$) measures the probability that objects that possess A will also contain*

$A \rightarrow B$. The rule is defined as follows:

$$conf(A \rightarrow B) = \frac{\mid (A \cup B)' \mid}{\mid A' \mid} \tag{2}$$

FCA Tools and Application Thereof Numerous tools and software packages have been developed over time to support FCA tasks and the creation of formal concept lattices [16, 17]. These instruments include Concept Explorer (ConExp), Galicia, Lattice Miner and FCART [14]. Ignatov [14] highlights the distinctions between these software alternatives, use-case advantages as well as limitations. Given the objectives of this paper and the documented advantages of each software ConExp was selected as the most appropriate and will therefore be discussed further.

ConExp ConExp was built in Java by S. Yevtushenko and has been widely endorsed due to its accessible FCA tools which facilitate essential techniques such as context editing, line diagram drawing, finding the Duquenne-Guigues base of implications, finding the base of association rules within a formal context, and performing attribute exploration [14]. Preference of ConExp also arises from the non-static nature of the resulting diagrams, which is favoured due to the interactivity it affords analysis like, for example, being able to move nodes [14]. Accordingly, ConExp can facilitate easier visualisations of the relationships between objects and attributes within the complete lattice, in addition to providing the necessary exploration tools.

FCA Application in Ontology Design During the last decade, FCA has been applied in a variety of fields for a variety of purposes [28]. In a two-part survey, [28] present an extensive overview of the materials published between the years 2003 and 2011 that utilise FCA for knowledge discovery and ontology engineering within numerous application domains. Findings showed that the majority of authors implemented FCA as a means to extract concept hierarchies originating from unstructured, semi-structured or structured texts. Data of interest was used to derive a concept lattice which thereafter was used to extract ontological classes of terms, a hierarchical ordering of concepts, as well as implications and associations between concepts. Finally, the resulting ontological knowledge was able to be encoded using an ontology language like OWL, with future texts being categorisable using the developed ontology. Additionally, OWL DL was used to encode the domain axioms for reasoning purposes.

2.4 Ontology Engineering

Semantic Technologies Semantic technologies use formal semantics to elucidate meaning from data in a manner that can be interpreted by machines. This is often achieved by the inclusion of axioms in an ontology to infer rules and constraints on resulting relationships.

Description Logics, Reasoning and Rules How knowledge is represented plays a vital role in semantic technologies [12]. Description logics (otherwise known as DL's) are a family of knowledge representation languages, derived from First Order Logic, that are widely used in ontological modelling [15].

Reasoning is a way to make sense of the ontological information by applying logic [12]. Inference is a type of reasoning that reaches a conclusion based on an existing premise. By combining an ontology with an inference engine, new, implicit knowledge can be derived from the explicit information via the application of inference rules (i.e. IF-THEN) [12]. A trade-off between expressability and decidability must be achieved via the implementation of rules in the form of description logic axioms [12]. There are separate languages for formulating rules, e.g. Semantic Web Rule Language.

Ontology Languages and Design Ontology engineering concerns the methodologies utilised for the design and implementation of an ontology [12]. When creating an ontology, it is important to maintain detailed knowledge about the domain of interest as well as agreed upon definitions of terms. Literature today posits several approaches for creating ontologies, with most approaches formulating requirements in the form of competency questions. Competency questions guide ontology development as the resulting ontology should be able to answer these questions in a manner that reveals new knowledge to the user.

Over the years, many languages for formulating ontologies have arisen, each with their own set of most appropriate use cases. Some of the more popular languages include Stanford's Knowledge Interchange Format (KIF) and the W3C's Resource Description Framework (RDF), Resource Description Framework Scheme (RDFS), Web Ontology Language (OWL) and OWL 2 (the revised version of OWL). The ontology language most suited for our purposes is the OWL 2 variant OWL 2 DL and it is for this reason that we will explore this language type further.

OWL 2 The Web Ontology Language known as OWL 2 was developed by the World Wide Web Consortium (W3C) as a means for representing ontologies specifically relating to the semantic web. There are many sub-languages under the OWL 2 umbrella, all with varying degrees of expressiveness. OWL 2 DL, being the variant with the most favourable computational properties, is the preferred language in the context of an FCA driven ontological infrastructure. Moreover, OWL 2 DL enjoys wider support from semantic reasoners than its counterparts due to its favourable computational properties [21]. A detailed explanation of the constructs that formulate OWL 2 DL is beyond the scope of this paper, however, insights in this regard can be obtained through the works of Grønberg [12].

2.5 Related Work

Aligning with the approach proposed by Obitko et al. [25], this paper sets out to follow the method set forth when developing an FCA ontology specific to

ICS cyberthreats. This methodology incorporates transforming an empty set of concepts and properties, to a populated cross table, to a visualised lattice formation using FCA tools. The process incorporates several rounds of editing so that modifications can be made to appropriately capture and analyse the data.

Moreover, [3] evaluate how FCA, knowledge discovery and artificial intelligence (AI) techniques can be used to better support security analysis in computer networks and systems. The road-map developed involves three axes namely: (1) using FCA to enrich vulnerability management techniques, (2) improving security knowledge representation for automated reasoning, and (3) enhancing cyber threat intelligence mechanisms. The short-term objective in (1) is to use FCA to study vulnerability anticipation mechanisms characterised by their properties, whereas (2) investigates the link between knowledge representation methods and security standard efforts such as structured threat information expression (STIX™), which is a language that can be used for the exchange of cyber security related data [23]. Ultimately, (3) integrates the results from (1) and (2) to develop improved approaches to deal with cyber-security threats. This helps develop a machine-assisted cyber threat analysis process that aids in detecting and recognising current and potential security attacks [3].

Foundational work at NATO by Parmar and Domingo has highlighted the need for additional visualisations and structures for ATT&CK [26]. Follow-up research by Watson et al. [35] uses FCA to explore a part of ATT&CK using FCA; that differs from this paper as it covers ATT&CK for mobile devices and does not consider the ontology aspect.

A thesis by Grønberg [12] has helped shed light into understanding the MITRE ATT&CK framework and adapting it to build a knowledge base for cyberthreat intelligence. The paper focuses on integrating the different information from MITRE's common attack enumeration and classification (CAPEC), a dictionary of known patterns of attack [19], and ATT&CK to build an ontology that would help users understand specific behaviours by threat actors under the enterprise domain. The resulting ontology has shown great potential as a tool for reasoning and modelling threats and actors to help enterprise system users prepare and defend their systems better [12].

3 Problem Statement and Research Objectives

The vulnerabilities of ICS can primarily be attributed to the growing connection of traditionally closed-circuit Industrial Controllers to the more widely connected enterprise systems of a company as well as ICS's reliance on microprocessor-based controllers. Although MITRE now provides an extensive knowledge base to combat adversary action through its ATT&CK framework for ICS, it fails to represent this knowledge in a navigable manner that reveals the deeper connections between adversary techniques and procedures, as well as in a way that prompts early intervention of adversary behaviours. Given the problem statement, this paper seeks to develop a proof of concept to demonstrate how FCA and ontology engineering can be applied to the ICS cyberthreat landscape in order to facilitate faster mitigated responses to cyber attacks. By encoding known adversary techniques and associated procedures into a formal concept lattice, FCA visualisation and analysis can be applied to reveal deeper associations and implications between the relevant software, which the matrix provided by MITRE is yet to achieve. Thereafter, by compiling the results of the FCA exploration to OWL 2 DL, machine interpretation and semantic web reasoning can be facilitated. The aforementioned intends to highlight related techniques as identified in ATT&CK, and apply taxonomic logic in order to help practitioners identify and anticipate which techniques are in use in their respective systems and which are likely to follow. Moreover, the query capability afforded by the ontology can assist security exports in the discovery of new knowledge. Accordingly, this paper aims to elevate the framework made available by MITRE by delivering a formal ICS cyberthreat ontology for the realisation of faster mitigated responses to cyber attacks. Thus, by combining FCA and ontology engineering, and using them complementarily, one is able to gain improved understanding of the relatively underdeveloped ICS threat landscape.

4 Methodology

4.1 Data Pre-processing

Firstly, the MITRE ATT&CK framework for ICS will be used to extract a data set of adversary techniques and associated software procedures. This data set consists of 81 techniques and 17 software and are referred to by their MITRE assigned IDs in order to simplify identification.

4.2 Data Mapping

This data will then be transformed into a formal context using a cross table, in which techniques are objects and software are attributes. In doing so, the appropriate software can be used to apply FCA tools for conceptual analysis. Using ConExp, a complete lattice will be constructed with the intention of highlighting the hierarchical ordering of techniques that results as well as the commonalities and variances between techniques on the basis of the software procedures utilised. Specific components of the resulting lattice will be highlighted by way of discussing any conspicuous techniques and procedures.

4.3 Data Exploration

Thereafter, using ConExp, the Duquenne-Guigues set of implications and association rules were calculated within the formal context. An in depth explanation of Duquenne-Guigues as a base is beyond the scope of this paper, however more information can be derived from Ignatov [14]. The results of this FCA exploration was documented and used to reveal the priority of attack techniques on the basis of the software linkages that are likely to trigger other types of adversary action within a system. Accordingly, software implications and associations categorised into probability groups according to whether they are low confidence (0.10-0.45), medium confidence (0.46-0.74) or high confidence (0.75-1.00).

The ontology structure was then populated with the ICS adversary techniques, tactics and other relevant information from the ICS ATT&CK framework using the Protégé ontology editor. This was achieved by clarifying the domain concepts that the ontology must cover, encoding the relevant classes to the ontology, populating the ontology with instances and specifying instance relations and finally, creating queries to test whether the ontology would be useful for representing information about adversary behaviour.

The domain of the ontology is adversary cyberthreat behaviours, which are classified into the following concepts: “Tactic”, “Technique”, “Software”, “Asset”, “Level”, “Group” and “Implication”. Tactic objects refer to an adversary’s strategy or intention and are related to associated techniques using the “applies” object property. Technique objects refer to the methods adversaries use to execute tactics, and are associated to a specific software implementation via the “usedBy” object property. Thus, a software “uses” a technique to achieve some objective (tactic). Software objects refer to the software procedures used to execute a technique and are associated to an ”Implication” as determined by the FCA analysis. Software objects can also be categorised based on their utility. Thus, “Malware” forms a sub-class of the software concept, which can be further classified into “Ransomware”, “Trojan”, “Virus” and “Worm” sub-classes respectively.

Moreover, a technique ”targets” an asset within an ICS. Thus, “Asset” refers to the hardware components affected by a technique. An asset object “belongsTo” a “Level”, as indicated by the Purdue model. Thus, “Level” articulates which level of the Purdue model is being targeted. Furthermore, a “Group” is a set of intrusion techniques referred to by a common name within the cybersecurity community. Thus, “Group” is associated to techniques and software via the “employs” object property. Software that are likely to be implemented successively are referred in the ontology as an “Implication” and are derived using FCA.

With the domain concepts clarified, technique, software, asset and implication instances are manually encoded to the ontological structure using the MITRE assigned IDs for techniques and software, and a generated ID for implications.

Implementing the resulting implications from the FCA as individuals rather than as general class axioms was chosen due to the fact that general class axioms generated invalid inferences. Thus, an “Implication” is modelled with the following properties, namely “hasPremise”, “hasConclusion” and “hasConfidence”, in order to associate the relevant software and likelihood of successive occurrence. By implementing “hasConfidence” as a functional data property, confidence based queries and general class axioms for confidence value binning based on the aforementioned ranges are added to the ontology for new knowledge discovery.

Lastly, reasoning and queries are applied to the resulting ontology, using HermiT. These queries aim to demonstrate how the resulting ontology can be used to infer new knowledge about threat adversaries. By using OWL 2 DL, the resulting ontology is able to be validated via the inferences made during reasoning. Accordingly, a formal ontology of adversary actions will emerge, as to achieve the stated objectives of this paper.

5 Data Analysis

5.1 Lattice Interpretation

Using the techniques and procedures identified in the ATT&CK framework for ICS as objects and attributes respectively, a formal context pertaining to ICS cyberthreats emerges in the form of a 1377 block matrix or cross table.

ConExp, which labels nodes according to the names of the techniques and procedures, was used to drill down on the specific techniques and produces within the complete lattice of the formal context. An initial glance at the complete lattice, as seen in Figure 1, reveals a concentration of implications on the left, middle to upper region of the lattice. These types of implications stand to reveal meaningful connections of how the threat procedures connect to one another, which stands to deepen one's understanding of how to combat such software.

Upwards edges between nodes encode logical implication which enables one to immediately derive more information about which techniques and software are likely to be used in combination, allowing for the faster mitigation of attacks. Thus, the lattice enables visualisation that allow insights that would be unrecognisable using the matrix provided by MITRE.

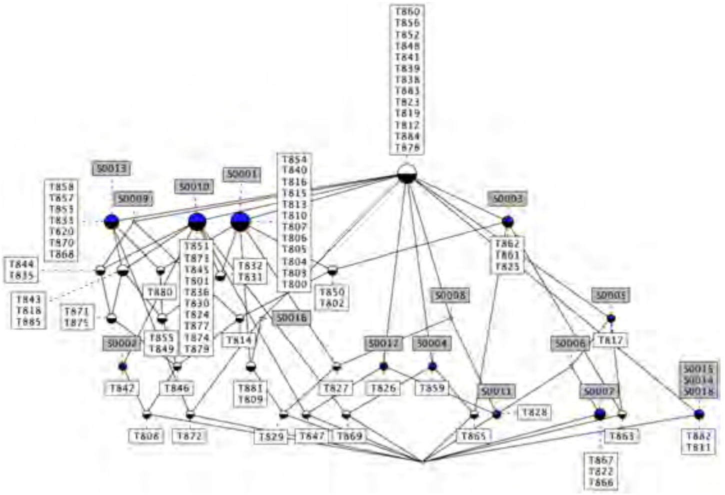


Fig. 1. ConExp Lattice showing the link between particular groupings of techniques and specific software's as determined by the MITRE ATT&CK Matrix for ICS. It reveals insight into which techniques contain which procedures and which techniques have procedures in common.

5.2 Implications and Associations

This section reports on the notable implications and associations between procedures as indicated in the above lattices. The report is based on the results from the solvers based on Duquenne-Guigues and Luxenburger bases.

An interesting relationship that is identified is between the procedures S0014, S0015 and S0018. The implication identified is S0018 → S0014, S0015 with two objects (techniques) supporting the implication. By referring at the MITRE framework procedure names, the implication can be translated to (ACAD/ Medre.A → Duqu, Flume). A closer look at the procedure descriptions reveals that S0018 is a worm that steals AutoCAD files with drawings, S0014 is a collection of malware that gathers intelligence data and assets, whereas S0015 is a worm that can open back doors and steal information such as AutoCAD drawings from a compromised computer. All these procedures work to steal and gather information for industrial espionage. The implication shows us that in a case where S0018 is used, S0014 and S0015 may be introduced.

Another relationship that we can look at is between the procedures S0009, S0010 and S0013. The implication identified is S0009, S0013 → S0010, with four out of four objects supporting the implication. The implication can be translated as PLC-Blaster, Triton → Stuxnet. The procedure descriptions reveal that S0009 is aimed at infecting PLC's, S0013 is a malware built to communicate with SIS controllers and S0010 is specially designed to target ICS devices. A look into their use cases reveals that they are all aimed at attacking PLC's. It is shown that the techniques they have in common are aimed at hunting for system vulnerabilities (T846), scanning targeted control devices for information (T808), loading executable program organisation units through a management API (T871) and halting the host PLC functions and executing malicious PLC code (T875). These techniques are found at the control level (Level 1) and they have the capability to impair access control into the host system, thus leaving the system at the adversary's mercy. By using the implications provided by the lattices, one can use them to quickly identify attacks and also be able to anticipate what the next moves would be and help setup better mitigation steps.

The confidence of association rules provide us with the probability that the implications will hold. The solvers required a minimum support and minimum confidence value which were provided as 0.01 and 0.10 respectively. The solver from ConExp revealed strict rules (equal to 1) and approximate rules (between 0.1 and 1). Due to the fact that there is a large number of associations, only a few are highlighted. The different associations are grouped according to the confidence values. The groups are low probability (0.10-0.45), medium probability (0.46-0.74) and high probability (0.75-1.00). The solvers show that the implication S0009, S0013 → S0010 has a confidence of association value of 1, which means there's a high probability that the techniques used by S0009 and S0013, will also be used by S0010. This shows that more attention would need to be paid to associations that have high probability values.

5.3 Ontology Queries and Inferences

In order to demonstrate how the ontology can be used to demonstrate new knowledge, a series of influential queries surrounding a hypothetical situation will now be discussed:

A security practitioner suspects a Data Historian asset is compromised within their system. The practitioner would first want to know which software would be involved in the attack which can be identified using the query

```
uses some (targets some DataHistorian)
```

This query returns fourteen software respectively. The practitioner would then want to know which implications follow from these software in order to identify which software are likely to follow from the initial attack and at what probability. It is presumed the practitioner is only concerned about implications with a high probability. Accordingly, this can be achieved using the query:

```
uses some (targets some DataHistorian) and isPremise some (
    hasProbability value "high")
```

Twelve software instances are returned. The practitioner is then able to identify the implications of interest by selecting the 'Explain inference' icon.

In the case of software S0001, twelve explanations detail that S0001 is a high probability premise of the following implications, namely IMP10, IMP11 and IMP21. The practitioner can now examine IMP10, IMP11 and IMP21 to find the software that are are likely to follow by using the query:

```
isConclusion some ({IMP10} or {IMP11} or {IMP21})
```

Four software are returned, namely S0003, S0009, S0010 and S0013. The practitioner can now identify the assets most likely to be affected by these software using the query:

```
usedBy some ({S0003} or {S0009} or {S0010} or {S0013}) and
    targets some (belongsTo some Level)
```

A list of Techniques used by these software is returned which can be used by the practitioner to identify at risk assets within the system by selecting the 'Explain inference' icons. Ultimately, the practitioner can implement countermeasures to mitigate or defer the attacks that follow from S0001 before they even occur. This process can be repeated for all high probability conclusions respectively.

6 Results and Discussion

The MITRE matrix for ICS is still relatively new and underdeveloped in nature, creating more room for some ambiguity and incomplete information. Using queries allows the user to assess relationships that extend further than relationships between techniques and concepts. The integration of confidence values takes the user a step further and allows the prioritisation of attacks through combining the efforts from the ontology and the information from the FCA results. The use of FCA associations strengthens the ontology and provides more certainty for the user. The frequency of Software and Techniques need not be queried in the ontology as the ontology's complementary tool, the FCA lattice, provides this information already. This kind of data is useful to a security specialist when deciding how to defend against ICS manipulations.

Due to the ontology's adoption of open world reasoning, which is defined as reasoning where the logic is not complete, some anomalies arose during the query process that significantly impacted the types of queries that could be run. Initial queries were much better suited to closed world reasoning, which is defined as reasoning where the logic is complete i.e. if something is proven false the opposite is true. To enforce closed world reasoning in this ontology would not have been realistic as Techniques, Implications, Confidence values etc. are all subject to change if new information were to arise.

7 Conclusions and Future Research

Emerging from this paper is an FCA exploration and ontology of ICS cyberthreats which stand to enrich the knowledge base provided by MITRE. By enriching the ATT&CK Matrix using an FCA driven ontological approach, value is established for cybersecurity specialist charged with protecting these systems. Both the FCA and ontology allow for a richer understanding of the threat landscape facing ICS, a relatively new domain of interest for cyberthreats.

Available threat models for ICS are currently relatively new. Accordingly, as the domain becomes more formalised, future iterations of this research should be able to leverage new information into the developed ontology. Similarly, extensions of the MITRE ATT&CK framework for ICS should be iteratively added to the formal context of the FCA analysis. In this regard, researchers have posited several approaches for the incremental construction of concept lattices [37].

Moreover, concerns of scalability in the context of FCA will need to be addressed. FCA applies well to the context of ICS because the domain of knowledge is limited. As the domain becomes more extensive, however, future iterations will also need to accommodate larger volumes of data which could prove challenging.

Furthermore, looking at the results obtained from the ontology as well as the challenges associated with assigning confidence levels in OWL, future research should focus on improved methods for incorporating probability values to ontological axioms. Accordingly, future iterations of this work should consider probabilistic knowledge bases via Bayesian networks as a means to represent FCA implications for probabilistic reasoning support [6, 7].

Acknowledgements We would like to thank Arina Britz for all her help on ontology structures and reasoning as well as our supervisors, Bruce W. Watson and Dewald Blaauw for their cybersecurity research guidance and support throughout.

References

1. The stride threat model (Nov 2009), https://docs.microsoft.com/en-us/previous-versions/commerce-server/ee823878(v=cs.20)?redirectedfrom=MSDN
2. Ackerman, P.: Industrial Cybersecurity. Packt Publishing Ltd, Birmingham (2017)
3. Barrère, M., Betarte, G., Codocedo, V., Rodríguez, M., Astudillo, H., Aliquintuy, M., Baliosian, J., Badonnel, R., Festor, O., Dos Santos, C.R.P., Nobre, J.C., Granville, L.Z., Napoli, A.: Machine-assisted cyber threat analysis using conceptual knowledge discovery - Position paper. CEUR Workshop Proceedings **1430**, 75–86 (2015)
4. Bechhofer, S., van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D.L., Patel-Schneider, P.F., Stein, L.A.: OWL Web Ontology Language Reference (2003)
5. Davis, R., Shrobe, H., Szolovits, P.: What Is Knowledge Representation? AI Magazine **14**(1), 17–33 (1993). https://doi.org/10.1609/aimag.v14i1.1029
6. Ding, Z., Peng, Y.: A probabilistic extension to ontology language OWL. Proceedings of the Hawaii International Conference on System Sciences **37**(C), 1775–1784 (2004). https://doi.org/10.1109/hicss.2004.1265290
7. Ding, Z., Peng, Y., Pan, R.: BayesOWL: Uncertainty modeling in semantic web ontologies. Studies in Fuzziness and Soft Computing **204**, 3–29 (2006). https://doi.org/10.1007/3-540-33473-4_1
8. Fenz, S., Heurix, J., Neubauer, T., Pechstein, F.: Current challenges in information security risk management. Information Management & Computer Security **22**(5), 410–430 (nov 2014). https://doi.org/10.1108/IMCS-07-2013-0053
9. Fransen, F., Smulders, A., Kerkdijk, R.: Cyber Security-Informationsaustausch zur Erkennung von Cyber-Bedrohungen und -Vorfällen. Elektrotechnik und Informationstechnik **132**(2), 106–112 (2015). https://doi.org/10.1007/s00502-015-0289-2
10. Ganter, B., Obiedkov, S.: Conceptual Exploration. Springer Berlin Heidelberg, Berlin, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49291-8
11. Graham, J., Hieb, J., Naber, J.: Improving cybersecurity for Industrial Control Systems. IEEE International Symposium on Industrial Electronics **2016-November**, 618–623 (2016). https://doi.org/10.1109/ISIE.2016.7744960
12. Grønberg, M.: An Ontology for Cyber Threat Intelligence. Ph.D. thesis, University of Oslo (2019)
13. Gruber, T.R.: Technical Report KSL 92-71 Revised April 1993 A Translation Approach to Portable Ontology Specifications by A Translation Approach to Portable Ontology Specifications. Knowledge Creation Diffusion Utilization **5**(April), 199–220 (1993). https://doi.org/http://dx.doi.org/10.1006/knac.1993.1008
14. Ignatov, D.I.: Introduction to Formal Concept Analysis and Its Applications in Information Retrieval and Related Fields. In: Braslavski, P., Karpov, N., Worring, M., Volkovich, Y., Ignatov, D.I. (eds.) Communications in Computer and Information Science, pp. 42–141. No. December 2015 in Communications in Computer and Information Science, Springer International Publishing, Cham (2015). https://doi.org/10.1007/978-3-319-25485-2_3
15. Krötzsch, M., Simancik, F., Horrocks, I.: A Description Logic Primer. CoRR (January 2012)
16. Kuznetsov, S., Watson, B.W.: Proceedings of Russian and South African Workshop on Knowledge Discovery Techniques Based on Formal Concept Analysis (RuZA15), vol. 1552. CEUR Workshop Proceedings (November 2015)
17. Kuznetsov, S., Watson, B.W.: Proceedings of the International Workshop on Formal Concept Analysis for Knowledge Discovery (FCA4KD), vol. 1921. CEUR Workshop Proceedings (2017)

18. McLaughlin, S., Konstantinou, C., Wang, X., Davi, L., Sadeghi, A.R., Maniatakos, M., Karri, R.: The Cybersecurity Landscape in Industrial Control Systems. Proceedings of the IEEE **104**(5), 1039–1057 (2016). https://doi.org/10.1109/JPROC.2015.2512235
19. MITRE: About CAPEC (2019), https://capec.mitre.org/about/index.html
20. MITRE: ATT&CK© for Industrial Control Systems (2020), https://collaborate.mitre.org/attackics/index.php/Main_Page
21. Motik, B., Grau, B.C., Horrocks, I., Wu, Z., Fokoue, A., Lutz, C., Calvanese, D., Carroll, J., Giacomo, G.D., Hendler, J., Herman, I., Parsia, B., Patel-schneider, P.F., Communications, N., Ruttenberg, A., Sattler, U.: OWL 2 Web Ontology Language Profiles (Second Edition) (2012), https://www.w3.org/TR/owl2-profiles/
22. Muckin, M., Fitch, S.C.: A Threat-Driven Approach to Cyber Security. Lockheed Martin pp. 1–45 (2014), http://www.lockheed.fi/content/dam/lockheed/data/isgs/documents/Threat-Driven Approach whitepaper.pdf
23. Oasis Open: Introduction to STIX (2020), https://oasis-open.github.io/cti-documentation/stix/intro.html
24. Obitko, M.: Ontologies - Description and Applications. Tech. Rep. April 2001, Czech Technical University in Prague (2001)
25. Obitko, M., Snásel, V., Jan, S.: Ontology Design with Formal Concept Analysis. CLA **128**(3), 1377–1390 (2004)
26. Parmar, M., Domingo, A.: On the Use of Cyber Threat Intelligence (CTI) in Support of Developing the Commander's Understanding of the Adversary. Proceedings - IEEE Military Communications Conference MILCOM **2019-November**, 1–6 (2019). https://doi.org/10.1109/MILCOM47813.2019.9020852
27. Pasquier, N.: Mining association rules using formal concept analysis. Contributions to the ICCS International Conference on Conceptual Structures **1867**(12), 259–264 (2000), http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.25.3360
28. Poelmans, J., Ignatov, D.I., Kuznetsov, S.O., Dedene, G.: Formal concept analysis in knowledge processing: A survey on applications. Expert Systems with Applications **40**(16), 6538–6560 (2013). https://doi.org/10.1016/j.eswa.2013.05.009
29. Rodin, D.N.: The cybersecurity partnership: A proposal for cyberthreat information sharing between contractors and the federal government. Public Contract Law Journal **44**(3), 505–528 (2015)
30. Sauerwein, C., Sillaber, C., Mussmann, A., Breu, R.: Threat Intelligence Sharing Platforms: An Exploratory Study of Software Vendors and Research Perspectives. In: Leimeister, J., Brenner, W. (eds.) 13th International Conference on Wirtschaftsinformatik. pp. 837–851. St. Gallen (2017)
31. Stouffer, K., Pillitteri, V., Lightman, S., Abrams, M., Hahn, A.: Guide to Industrial Control Systems (ICS) Security NIST Special Publication 800-82 Revision 2. NIST Special Publication 800-82 rev 2 pp. 1–157 (2015). https://doi.org/10.6028/NIST.SP.800-82r1
32. Strom, B.E., Miller, D.P., Nickels, K.C., Pennington, A.G., Thomas, C.B.: MITRE ATT&CK ™ : Design and Philosophy. Tech. Rep. July, The MITRE Corporation (2018), https://www.mitre.org/sites/default/files/publications/pr-18-0944-11-mitre-attack-design-and-philosophy.pdf
33. Stumme, G.: Formal Concept Analysis on Its Way from Mathematics to Computer Science. In: Priss, U., Corbett, D., Angelova, G. (eds.) Conceptual Structures: Integration and Interfaces, pp. 2–19. Springer,Berlin, Heidelberg (2002). https://doi.org/10.1007/3-540-45483-7_2

34. VERIS: The vocabulary for event recording and incident sharing (2020), http://veriscommunity.net/index.html
35. Watson, L.R., Watson, B.W.: Exploratory data science on ATT&CK. In: submitted to: CyCon 2021. NATO Cooperative Cyber Defence Centre of Excellence (2020)
36. Wille, R.: Conceptual Landscapes of Knowledge: A Pragmatic Paradigm for Knowledge Processing. In: Classification in the Information Age, pp. 344–356. Springer,Berlin, Heidelberg (1999). https://doi.org/10.1007/978-3-642-60187-3_36
37. Zhang, J., Liu, R., Zou, L., Zeng, L.: A new rapid incremental algorithm for constructing concept lattices. Information (Switzerland) **10**(2), 1–12 (2019). https://doi.org/10.3390/info10020078

Exploiting a Multilingual Semantic Machine Translation Architecture for Knowledge Representation of Patient Data for Covid-19

Laurette Marais[1] and Laurette Pretorius[2]

[1] Digital Audio and Visual Technologies, Next Generation Enterprises and Institutions, CSIR, Pretoria, South Africa
laurette.p@gmail.com
[2] College of Graduate Studies, University of South Africa, Pretoria, South Africa
laurette@acm.org

Abstract. AwezaMed Covid-19 is a multilingual speech-to-speech translation application for screening patients for Covid-19. It enables English-speaking health care providers (HCPs) to conduct screenings by asking questions to patients in all other official languages of South-Africa. It uses a multimodal computational grammar translation system to enable English speech and screen-based input, which can be translated to produce synthetic speech in the target languages. Grammatical Framework is used for the translation system, utilising a semantic interlingua. Because of this, each utterance translated by the application is represented by a semantic tree, which could be exploited for knowledge representation.

openEHR is an open standard specification for electronic health records, where the goal is to enable interoperability between patient record systems. A central feature is the definition of archetypes and templates, which are formal models of clinical content. Various archetypes and templates focused on knowledge related to Covid-19 have been developed in recent months.

In this paper, we describe how the machine translation architecture designed for multilingual speech-to-speech translation can be adapted for knowledge representation consistent with existing Covid-19 openEHR archetypes, which could be recorded seamlessly by HCPs during the screening.

1 Introduction

South Africa is a multilingual society with 11 official languages. English serves as a lingua franca in many spheres and it is the language in which most health care providers (HCPs) are educated at tertiary level. However, large numbers of South Africans are not proficient in English[3], creating a language barrier to health care in many settings. At hospitals and clinics, it is often the case that

[3] Stats SA language use...

other staff, including security guards and cleaners, are called upon to interpret in cases where the HCP and patient do not share a common language. This has a detrimental effect on efficient use of staff, patient privacy and the ability of HCPs to provide respectful care.

The Covid-19 pandemic has focused attention on the ability of health care systems to cope with enormous amounts of patients at once, as well as the need to record data that may aid in understanding and responding to dangerous viral outbreaks. Both these factors are severely affected if language barriers exist between HCPs and patients.

To enable communication for screening patients for Covid-19, a speech-to-speech mobile translation application was developed to translate spoken English utterances to spoken Afrikaans, isiXhosa and isiZulu utterances. AwezaMed Covid-19 uses automatic speech recognition (ASR), grammar-based machine translation (MT) and text-to-speech (TTS) to enable communication by an HCP to a patient. The application was adapted from a different version of AwezaMed geared towards maternal health.

In this paper, we describe how AwezaMed-Covid-19 can be adapted for knowledge representation as Linked Data, consistent with existing Covid-19 openEHR archetypes, which could be recorded seamlessly by HCPs during the screening. The structure is as follows: We provide a brief overview of related work in Section 2. Two major factors contributed to the choice to provide translations in only one direction, namely from the HCP to the patient. We discuss them in Section 3. In Section 4 we describe the multilingual speech-to-speech translation system, and in Section 5, we develop the main argument of this paper, namely that the speech-to-speech architecture chosen due to the constraints provided by the use case, enables knowledge representation of patient responses. A straight-forward extension to the application would enable HCPs to capture these responses with a single click, with the responses stored in the form of RDF triples that could be queried using a language such as SPARQL. The application would then be able to assist HCPs to screen patients who are not proficient in English, as well as allow fast capturing of knowledge that would be useful for epidemiological purposes.

2 Contextualisation

The amount of patient data, including Covid-19 information, is constantly increasing world wide and there is an urgent need to have a clear picture of the development and spread of the pandemic, also in developing countries such as South Africa. Indeed, the rapid acquisition, publication and interoperability of such data have a high priority. In the past three decades various standards, vocabularies and knowledge representations have been developed for this purpose, for example, ISO standards [4], SNOMED-CT [5], openEHR [6], HL7, [7] and FHIR. [8] This resulted in patient information [12] encoded in a variety of formalisms (knowledge representations).

openEHR is a technology for creating and managing electronic health care records (EHRs), "consisting of open specifications, clinical models and software that can be used to create standards, and build information and interoperability solutions for healthcare"[9]. An important aspect of how openEHR deals with knowledge is its two-level model[4], consisting of the reference model (RM), which defines models of information content, such as data types and data structures, and the archetype model (AM), in which more specific domain knowledge can be described, such as the results of a laboratory test. The RM is stable and implemented in software,[10] while the goal of the AM is to allow clinicians to develop formal models that reflect their domain knowledge[7]. The Clinical Knowledge Manager (CKM) is a platform for sharing and collaborating on such domain knowledge models in the form of *archetypes* and *templates*.

In recent months, an archetype for Covid-19 symptoms[11] was contributed, in the form of a specialisation of the existing archetype for symptoms.[12] A template for Covid-19 was also developed[8], which includes archetypes for assessments, clinical background, treatment and discharge, and is available in the CKM.[13]

In openEHR, archetypes typically model healthcare concepts, such as blood pressure, while templates typically model forms, documents and messages. Archetypes are defined in ADL (Archetype Description Language) and can be queried using AQL (Archetype Query Language), analogous to how SPARQL can be used to query RDF (Resource Description Framework) triple stores. An essential difference between using openEHR and RDF to represent clinical knowledge, is the kinds of system interoperability they aim at. openEHR intends to enable interoperability in health care systems with a focus on electronic health care records, whereas RDF is domain independent and intended to be used at web scale.

The Semantic Web can be thought of as a suite of semantic web technologies together with Linked Data, a set of best practices for sharing data on the web. These semantic web technologies allow the creation of data stores on the web, the building of vocabularies, and the writing rules for handling data. Linked Data are supported by technologies such as RDF, SPARQL, OWL, and SKOS [14]. In RDF,

[4] https://www.iso.org/files/live/sites/isoorg/files/store/en/PUB100343.pdf

[5] https://www.snomed.org/snomed-ct/five-step-briefing

[6] https://www.openehr.org/

[7] https://www.hl7.org/

[8] http://hl7.org/fhir/

[9] https://www.openehr.org/about/what_is_openehr

[10] https://specifications.openehr.org/releases/BASE/latest/architecture_overview.html

[11] https://ckm.openehr.org/ckm/archetypes/1013.1.4399

[12] https://ckm.openehr.org/ckm/archetypes/1013.1.195

[13] https://ckm.openehr.org/ckm/templates/1013.26.291

[14] https://www.w3.org/standards/semanticweb/

a description of a resource [15] is represented as a number of triples, each of which codifies a statement about semantic data and consists of a *subject*, *predicate* and *object* [3] [1]. These subjects, predicates and often also the objects[16] themselves are URIs of concepts that reside in precise formal vocabularies and ontologies. RDF, therefore, relies on *semantics by reference*[17] . As the abstract data model of the Semantic Web, RDF is considered one of the dominant graph technologies currently driving semantic computing over web-scale distributed data.

More specifically, semantic web technologies and Linked Data, combined with big data analytics, have become key to making patient data semantically interoperable and to helping create predictive models on how epidemics might spread around the world [18] [12]. It is therefore important to ensure that patient data, also for Covid-19, is exposed as Linked Data as accurately and as seamlessly as possible.

Grammatical Framework (GF) [15] is a programming language for grammar engineering, which uses a semantic interlingua for multilingual machine translation. It has become the *de facto* standard in multilingual controlled natural language applications [16]. GF has been used in a number of knowledge representation projects. For example GFMed [10, 17] is a question answering system for biomedical interlinked data. It employs GF grammars for a controlled language targeted towards biomedial information and the SPARQL query language. In [2] an approach to multilingual ontology verbalisation of controlled language based on GF and the *lemon model* [19] is presented.

A number of knowledge representation projects in the indigenous South African languages have been reported. For example, [5, 6] discuss isiZulu verbalisation patterns for basic logic constructs (quantification, subsumption, conjugation, and negation) and devised algorithms to generate grammatically correct isiZulu sentences.

To the best of our knowledge, AwezaMed Covid-19 is unique in that it is the only speech-enabled mobile application currently in existence that supports multilingual, multimodal machine translation to all South African languages based on a semantic interlingua. This paper describes how knowledge representation can be added.

3 The speech-to-speech translation architecture

In this section, we discuss the choice of a semantic interlingua translation architecture for our specific use case, namely facilitating screening of patients for Covid-19, and the effect this has on the way the application can facilitate multilingual communication between HCPs and patients.

[15] A web resource, or simply resource, is any identifiable thing, whether digital, physical, or abstract. Resources are identified using Uniform Resource Identifiers (URIs)

[16] The object can also be a literal

[17] https://www.w3.org/TR/rdf-mt/

[18] https://www.ontotext.com/blog/linked-data-solutions-in-healthcare/

[19] https://lemon-model.net/

3.1 Choosing a suitable translation architecture

Deep learning techniques have made enormous progress in the last few years and are the state-of-the-art in machine translation for the large languages of the world. They rely on the very large amounts of data that have become available for many languages. However, all the official South African languages excluding English are so-called under-resourced languages, which means that the same amounts of data are not available – and in many cases do not exist – as for the larger languages of the world. This means that other techniques are worth investigating if a specific use case could benefit from it.

Available MT capability for English to other South African languages The health care domain requires a high level of accuracy in machine translation. Achieving appropriate accuracy using state-of-the-art data-driven machine translation techniques depends on two major factors: the availability of domain appropriate parallel data for each language pair, and the linguistic similarity between the source and target languages. [11] report BLEU scores for translation from English to isiXhosa and isiZulu of 37.11 and 44.07, respectively. However, these scores are the result of training and testing on the JW300 corpus, which contains hundreds of thousands of sentences trawled from JW.ORG, and therefore covers only a single domain. When the same systems were tested on the Autshumato test corpus, which consists of data from government documents [13], BLEU scores dropped dramatically to 1.42 and 1.56, respectively.

While data-driven approaches for these language pairs have some way to go to achieve high accuracy machine translation in domains where data is scarce, such as health care, a grammar-based approach can provide a more controlled form of domain appropriate translation, where coverage is constrained, but a high level of accuracy is guaranteed.

The goal of GF is reduce the amount of effort and time traditionally required by rule-based machine translation[15], so that building multilingual domain specific machine translation systems for real world applications is feasible. A particularly attractive aspect of using a rule-based approach such as GF is the ability to directly fix bugs, as well as other behaviour that may not be wrong, but could be inappropriate for a specific use case.

Speech-to-speech and mobile application integration GF as a programming language is well-suited to supporting multimodal, multilingual applications. Its diverse module system makes it possible to linearise the same utterance in different languages, as well as different formats for the same language. For example, in order to accept input from the ASR component, a grammar that parses lowercase, non-punctuated text is necessary, but text that can be displayed on a screen and fed to the TTS component must be punctuated and capitalised correctly.

Given the way a grammar limits the domain of supported utterances, it must support a touch input modality that can present the translatable content of the application to the user in a compact and intuitive way. This acts both as a mechanism for familiarising users with the domain covered by the application, as well as a fallback mechanism for when speech recognition fails.

Different roles of participants in a screening The nature of the utterances used by HCPs during a screening is relatively structured and predictable. This is especially true when the domain is limited to a subdomain of health care, such as screening for Covid-19. The grammars in AwezaMed were developed in close consultation with various HCPs to cover relevant and useful content.

In contrast to HCPs, who have specific domain knowledge and are responsible for driving the communication during a screening in order to arrive at a correct finding, patients are not domain experts. Their answers to questions arise from their experiential knowledge of their health, and may range widely in terms of detail, focus and applicability. Hence, constrained domain grammars are not suitable to model typical patient utterances.

Furthermore, repeated use of the application by the HCP will improve familiarity with the coverage of the application, ensuring more effective use over time. This is clearly not the case for patients, who in many cases will be using the application for the first time.

3.2 A communication model for grammar-based machine translation

Limiting speech-to-speech translation to only cover utterances uttered by the HCP has an obvious impact on the kinds of utterances that will enable HCPs to conduct an entire screening using the application. Specifically, HCPs will have to understand the patient responses without the help of the application. Hence, the communication model of the AwezaMed application limits almost all utterances to binary questions (i.e. requiring only a “yes” or “no” answer from the patient). In reality, therefore, the content of the application is such that statements of fact are presented to the patient in the form of questions which they can confirm or deny. The HCP need only understand the words for “yes” and “no” in the patient’s language, or understand gestures such as nodding or shaking of the head, to establish relevant observations about the patient’s health.

We see, therefore, how an analysis of the specifics of the use case leads to the choice of architecture and communication model, namely a GF-based domain grammar that translates binary questions posed by the HCP in the source language to a target language that the patient understands.

4 Semantic interlingua machine translation for Covid-19 screening

We turn now to the mechanism employed by GF to support domain grammars, namely a semantic interlingua architecture. The goal of a GF domain grammar, known as an application grammar, is to start with the semantics of the domain, and express it in one or more natural languages[14]. The semantics is defined via categories and functions in an abstract syntax, while one or more concrete syntaxes define how such categories and functions are linearised as strings.

Developing a concrete syntax therefore involves defining a *linearisation category* for each category in the abstract syntax, and a *linearisation function* for each function in the abstract syntax. Parsing is based on inversion of the linearisation rules in a non-trivial way[14]. GF can therefore be seen as a multi-source, multi-target compiler[9], where any string in one of the concrete syntaxes of the grammar can be translated to any other language by parsing the source utterance string into an abstract syntax tree and linearising the tree into a string in the target language.

The way in which semantically equivalent translation is achieved can be understood by considering Wittgenstein's notion of a language game[14]. An application grammar is effectively the definition of a specific language game, where translation is possible if the same language game can be played in both the source and target languages. Stated differently, starting with the semantics of a domain, translation is possible if the same abstract syntax tree, capturing some meaning in a specific domain, can be expressed as natural language utterances in two or more languages. Because translation is achieved via a semantic interlingua in the form of the abstract syntax, as long as the utterance in the source language represents the intended meaning of the user in the context of the domain, the user can be confident that the translation of the utterance represents the same meaning in the target language.

4.1 Implementing multilingual, multimodal machine translation for a mobile application

The application contains four application grammars, namely *Symptoms*, for asking about symptoms associated with Covid-19, *Medical History*, for establishing the presence of possible comorbid conditions, *General History*, covering allergies and substance usage, and *Covid-19*, which is mainly for relaying information and instructions related to Covid-19. For the purpose of showing how the grammar enables translation via a semantic interlingua, our focus will be on the *Symptoms* grammar.

Semantic trees in the abstract syntax The *Symptoms* application grammar allows the HCP to ask questions about Covid-19 related symptoms, including whether a patient has a certain symptom, whether the symptom started more, less or about a certain number of days ago, whether the symptom is persistent and whether the symptom is worsening. Fig. 1 shows an example of an abstract syntax tree, which is expressed in English as "Did the fatigue start about two days ago?". In each node label, the function name used to construct that particular constituent appears to the left of the colon, while the category type of the constituent appears to the right.

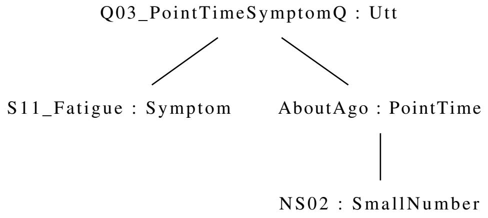


Fig. 1: Semantic tree for "Did the fatigue start about two days ago?"

Multiple languages and modes Each concrete syntax provides rules for linearising semantic trees to strings. This involves defining how every category in the grammar is represented as a record, namely its *linearisation category*, as well as how each function is applied to the linearisation categories, namely its *linearisation function*. A detailed discussion of how GF provides parameters and tables to implement grammar rules is beyond the scope of this paper, and the interested reader is referred to [15]. It suffices to say that the implementation of a concrete syntax for a specific language enables the GF runtime to generate strings in that language in a compositional way to represent the meaning of an abstract syntax tree.

The difference between the way the tree in Fig. 1 is expressed in English and isiZulu can be seen by comparing Fig. 2 and Fig. 3, which show how the semantic constituents of the abstract syntax tree are linearised into strings in the two languages. In order to facilitate the discussion in Section 5, the strings are colour coded, with strings in violet contributed by the `Symptom` category, green by the `PointTime` category, BurntOrange by the `SmallNumber` and black by the `Utt` category, which is the start category.

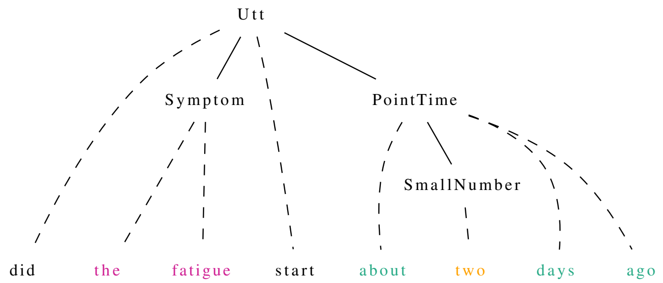


Fig. 2: Parse tree for expressing "Did the fatigue start about two days ago?" with ASR formatting conventions

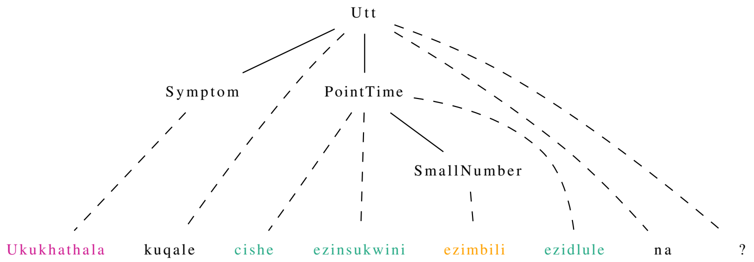


Fig. 3: Parse tree for expressing "Did the fatigue start about two days ago?" in isiZulu

Grammar-driven dynamic screen Note that the English string in Fig. 2 contains no punctuation or capitalisation. This is because the grammar contains a concrete syntax specifically for defining the appropriate English strings with formatting that follows the conventions of the automatic speech recognition component of the application. For the isiZulu strings, which must be displayed on the screen and also serve as input to the text-to-speech component, capitalisation and punctuation must be included.

However, in addition to the speech modalities supported by the grammar, the touch modality must also be supported. To this end, another version of the English concrete syntax exists, which adds markup to a capitalised and punctuated English string, which can be used to determine which parts of the string must be "live", in the sense that the user could click on it to change it. Fig. 2 shows the parse tree for the example utterance generated by this concrete syntax (slightly simplified for readability), while Fig. 5 shows how the user interface uses the information encoded in the marked up string. Each function in the grammar that produces an `Utt`, which is the start category of the grammar, corresponds to a so-called *dynamic utterance*, which could be thought of as a dynamic, grammar driven template for presenting many utterances on a single screen in an intuitive way.

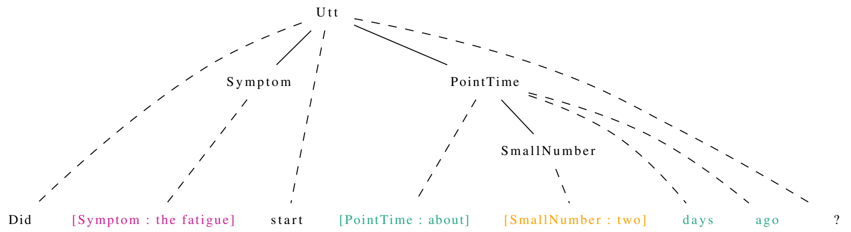
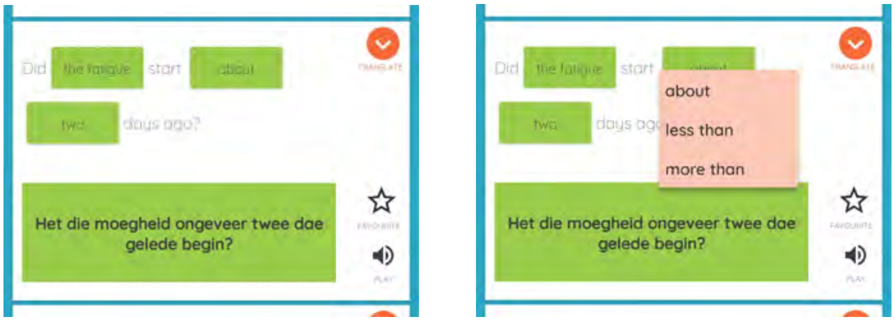


Fig. 4: Slightly simplified parse tree for expressing "Did the fatigue start about two days ago?" with dynamic utterance markup



Did the fatigue start about two days ago?

Het die moegheid ongeveer twee dae gelede begin?

Did the fatigue start two days ago?

about

less than

more than

Het die moegheid ongeveer twee dae gelede begin?

(a) Dynamic utterance (b) Making a selection

Fig. 5: Screen elements derived from dynamic utterance markup

5 Knowledge representation of patient responses

How does this speech-to-speech architecture, chosen due to the constraints provided by the use case, enable knowledge representation of patient responses?

5.1 From questions to observations

The translation architecture uses an interlingua that represents utterances as semantic trees. A binary question is, in effect, a statement put to one's interlocutor in the form of a question, which can then be confirmed or denied. When a question is put to the patient, and the patient answers in the affirmative, the semantic tree effectively represents an observation about the patient. For example, if the question "Did the fatigue start about two days ago?" is confirmed by the patient, the statement "The patient reports fatigue, which started two days ago" could be noted as an observation.

Practically speaking, the HCP might use the application to input some binary question in English which the app would then translate to the appropriate target language. When the patient answers in the affirmative, the HCP would want to capture this information in an electronic health record. In order to do this, the app should be extended with two specific features:

- Implementation of a patient information section for creating and updating patient entities in a suitable data repository. This would allow each screening session to add the observations obtained to a specific patient's health record.
- Implementation of a check box next to each dynamic utterance in the application, which would allow HCPs to immediately mark some utterance as being confirmed by the patient.

The first features concerns the management of patient EHRs in healthcare systems. Our contribution relates to the second feature, which would allow capturing of clinical observations in a seamless and semantically reliable way.

5.2 Generating knowledge representations with GF

In openEHR, an OBSERVATION is a type of ENTRY suitable for symptoms, test results and other clinical information that is, essentially, uninterpreted. The other entry types are ADMIN_ENTRY, EVALUATION, INSTRUCTION and ACTION. In this section we show how the *Symptoms* grammar can be extended to generate knowledge representations in the form of openEHR OBSERVATIONs, as well as RDF triples that similarly represent the observation.

Since our goal is to describe the mechanism by which a semantic interlingua can be used to express the confirmation of natural language binary questions as formal knowledge, we focus on generating just those snippets of code in both formalisms. Representing administrative information, that would connect these observations to patients is outside the scope of this paper.

Representing knowledge according to openEHR specifications can be done in JSON, and similarly, RDF triples can be encoded using Turtle. In order to generate JSON and Turtle code that represent observations, we added two concrete syntaxes to the existing translation architecture. Therefore, in addition to multiple languages and speech modalities, the grammar was extended to support two formal knowledge representation formalisms.

In the case of openEHR, the JSON code generated by the grammar would be integrated into a COMPOSITION structure that includes all necessary administrative information, while the RDF triples generated by the grammar can be added to a triple store that similarly connects the observation to the relevant administrative information.

As for any other concrete syntax, linearisation categories and linearisation functions were be defined in order to generated JSON and Turtle code. Given the context-free nature of JSON and Turtle, records containing a single string sufficed as linearisation categories, and straight-forward token concatenation was employed in the linearisation functions. Fig. 6 and Fig. 7 show strings generated by the two new concrete syntaxes, with the same colour coding as before.

The openEHR JSON concrete syntax generates code consistent with the Covid-19 symptom archetype, and uses the SNOMED-CT vocabulary to refer to specific symptoms. The RDF Turtle concrete syntax uses SNOMED-CT in the same way, in addition to the FHIR ontology.

In both code snippets, the string `<_SYSDATE_ - 0000-00-02>` is generated by the grammar with the intent that the host application resolve this based on the system time of the mobile device. The host application must also generate a unique identification number for each observation. Due to the differences in how observations are included in the different data repositories, the Turtle snippet includes the string `_OBSERVATION_ID_` that must be replaced. In the case of the openEHR JSON code, this string occurs in a different part of the `COMPOSITION` data structure, and is therefore not shown here.

The reason that the `onset_time` concept in the Turtle code is defined separately as shown, is because RDF provides more freedom to define time related concepts, via the W3C Time ontology,[20] than is possible given the Covid-19

[20] https://www.w3.org/TR/owl-time/

```
{ "content":
  { "archetype_node_id": "openEHR-EHR-CLUSTER.symptom_sign-cvid.v0",
    "type": "OBSERVATION",
    "name": { "value": "Covid-19 symptom" },
    "archetype_details":
    { "archetype_id": { "value": "openEHR-EHR-CLUSTER.symptom_sign-cvid.v0" } },
    "data":
    { "archetype_node_id": "at0001",
      "type": "ITEM_TREE",
      "name": { "value": "components" },
      "items": [
      { "archetype_node_id": "at0001.1",
        "type": "ELEMENT",
        "name": { "value": "Symptom/Sign name" },
        "value":
        { "type": "DV_CODED_TEXT",
          "value": "Fatigue",
          "defining_code":
          { "terminology_id": { "value": "SNOMED-CT" },
            "code_string": "84229001" } } },
    { "archetype_node_id": "at0152",
      "type": "ELEMENT",
      "name": { "value": "Episode onset" },
      "value":
      { "type": "DV_DATE_TIME",
        "value": "<_SYSDATE_ - 0000-00-02>" } } ] } } }
```

Fig. 6: Generated JSON snippet of the tree in Fig. 1

```
@base <http://example.org/> .
@prefix rel: <http://www.perceive.net/schemas/relationship/> .

@prefix fhir: <http://hl7.org/fhir/> .
@prefix time: <http://www.w3.org/2006/time#> .
@prefix sct: <http://snomed.info/id/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<_OBSERVATION_ID_>
    a [ fhir:ClinicalImpression sct:84229001 ] ;
    sct:405795006 <onset_time> .

<onset_time>
    a time:Instant .
    time:hasTime "<_SYSDATE_ - 0000-00-02>"^^xsd:dateTime .
```

Fig. 7: Generated Turtle snippet of the tree in Fig. 1

symptom archetype available in openEHR. In the latter, the symptom onset can be given as a time instant of type DV_DATETIME. However, the Time ontology allows more complex time concepts, with properties `time:before` and `time:after`. These properties can be used to represent binary questions in the grammar such as “Did the fatigue start more than two days ago?” and “Did the fatigue start less than two days ago?”. The latter example is represented as follows, and is generated when the `PointTime` constituent is created using the `LessThanAgo` function instead of the `AboutAgo` function:

```
<onset_time>
    a time:Instant .
    time:after "<_SYSDATE_ - 0000-00-02>"^^xsd:dateTime .
```

Depending on the choice of knowledge representation formalism, the application could disable check boxes for any instances of dynamic utterances it cannot represent. Another solution might be for a clinician to contribute an openEHR archetype for symptoms that contains additional item elements for expressing relative time concepts.

6 Discussion

Electronically capturing the observation example used throughout the paper, namely “The patient reports fatigue, which started two days ago”, would typically require that an HCP select the symptom, fatigue, from a drop down box on a screen, and select the date of two days ago from a calendar widget. This is time consuming and uncomfortable, because it interferes significantly with the interaction between the HCP and patient. Alternatively, the HCP must write the information down in order to capture it later, which is an inefficient and error prone process.

The essence of our contribution is in having outlined the theory and implemented the core components that would enable a seamless process for capturing patient information, even in cases where the HCP and patient do not share a common language. Relatively complex data can be captured by speaking it as a question, letting the application translate the input to speech in a different language, and checking a box when it is confirmed. In this way, we let the communication act facilitated by the mobile application do the heavy lifting of establishing the semantic content of the information to be captured.

For the sake of clarity, we have used a single example throughout the paper, but in reality, our implementation covers 396 unique utterances. We have implemented four utterance types, which reference 12 Covid-19 related symptoms, three different ways of expressing relative time, and 10 small numbers for counting days. In implementing the knowledge representation concrete syntaxes to support these utterances, concepts and formal information models that correspond to the utterances were identified within the openEHR and HL7 frameworks. This includes the openEHR archetype for Covid-19 symptoms, the SNOMED-CT vocabulary and several other ontologies.

In order to transform the existing AwezaMed app into a knowledge representation aid for Covid-19 screening as described in this paper, the steps identified in [12] could be implemented in two phases:

- **Ontology Development** Other utterances supported by the grammar must be analysed and the concepts they express must be associated where possible with existing knowledge representation concepts. In cases where the necessary terms, classes, properties and constraints do not exist, ontology development is required.
- **Semantic Data Creation** This phase entails the extension of the application with the two features mentioned in Section 5.1, as well as its integration with real data repositories containing semantic data of real patients.

7 Conclusion

The starting point of this paper was an existing speech-to-speech translation application, which was implemented using a semantic interlingua, chosen due to the constraints of the use case. We extended this architecture to provide a way to formally represent the knowledge gained while using the application.

This extension was implemented as additional concrete syntaxes in the application's GF translation system: in addition to linearising (or parsing) a semantic tree as a binary question in multiple languages, it was also linearised by the grammar into two knowledge representation formalisms.[21]

[21] The natural language formulation of the observation, namely "The patient reports fatigue, which started about two days ago" could just as easily be linearised by an additional concrete syntax, resulting in human readable statements about the observations made during the screening. In fact, this could be done in a multilingual way to make the content available as natural language in all official South African languages.

By implementing the extension according to established formalisms, namely openEHR and RDF, which exist within larger frameworks for representing knowledge in health care systems and on the web, we have shown how an application such as AwezaMed could integrate with such systems to contribute to the acquisition, publication and interoperability of health care information. This, in turn, would serve to enable better understanding and improved responses to viral pandemics such as Covid-19.

References

1. W3C RDF 1.1 Primer (2014)
2. Davis, B., Enache, R., Van Grondelle, J., Pretorius, L.: Multilingual verbalisation of modular ontologies using gf and lemon. vol. 7427 (08 2012)
3. Heath, T., Bizer, C.: Linked Data: Evolving the Web into a Global Data Space. Morgan & Claypool (2011), iSBN: 9781608454303
4. Kalra, D., Beale, T., Heard, S.: The openehr foundation. Studies in health technology and informatics **115**, 153–173 (2005)
5. Keet, C., Khumalo, L.: Toward verbalizing ontologies in isizulu. In: Davis, B., Kaljurand, K., Kuhn, T. (eds.) Controlled Natural Language. Springer International Publishing, Cham (2014), iSBN: 9783319102238
6. Keet, M., Khumalo, L.: Toward a knowledge-to-text controlled natural language of isizulu. Language Resources and Evaluation **51**(1), 131–157 (Mar 2017)
7. Leslie, H., Heard, S., Garde, S., McNicoll, I.: Engaging clinicians in clinical content: herding cats or piece of cake? In: MIE. pp. 125–129. Citeseer (2009)
8. Li, M., Leslie, H., Qi, B., Nan, S., Feng, H., Cai, H., Lu, X., Duan, H.: Development of an openehr template for covid-19 based on clinical guidelines. J Med Internet Res **22**(6), e20239 (Jun 2020). https://doi.org/10.2196/20239, http://www.jmir.org/2020/6/e20239/
9. Listenmaa, I.: Formal Methods for Testing Grammars. Ph.D. thesis, Gothenburg, Sweden (2019)
10. Marginean, A.: Question answering over biomedical linked data with grammatical framework. In: Cappellato, L., Ferro, N., Halvey, M., Kraaij, W. (eds.) Working Notes for CLEF 2014 Conference. Sheffield, United Kingdom (Sep 2014), https://hal.inria.fr/hal-01086472
11. Martinus, L., Webster, J., Moonsamy, J., Jnr, M.S., Moosa, R., Fairon, R.: Neural machine translation for South Africa's official languages (2020)
12. McGlinn, K., Hussey, P.: An Analysis of Demographic Data in Irish Healthcare Domain to Support Semantic Uplift. In: ICCS 2020, LNCS 12140 (2020)
13. McKellar, C.A., Puttkammer, M.J.: Dataset for comparable evaluation of machine translation between 11 South African languages. Data in Brief **29**, 105146 (2020). https://doi.org/https://doi.org/10.1016/j.dib.2020.105146, http://www.sciencedirect.com/science/article/pii/S2352340920300408
14. Ranta, A.: Type theory and universal grammar. Philosophia Scientiæ. Travaux d'histoire et de philosophie des sciences (CS 6), 115–131 (2006)
15. Ranta, A.: Grammatical Framework: Programming with Multilingual Grammars. CSLI Publications, Stanford (2011), iSBN-10: 1-57586-626-9 (Paper), 1-57586-627-7 (Cloth)

16. Safwat, H., Davis, B.: A brief state of the art of CNLs for ontology authoring. In: Davis, B., Kaljurand, K., Kuhn, T. (eds.) Controlled Natural Language. pp. 190–200. Springer International Publishing, Cham (2014), iSBN: 9783319102238
17. Unger, C., Forascu, C., Lopez, V., Ngonga Ngomo, A.C., Cabrio, E., Cimiano, P., Walter, S.: Question Answering over Linked Data (QALD-4). In: Cappellato, L., Ferro, N., Halvey, M., Kraaij, W. (eds.) Working Notes for CLEF 2014 Conference. Sheffield, United Kingdom (Sep 2014), https://hal.inria.fr/hal-01086472

Part V

Machine Learning Theory

Machine Learning Theory: Abstracts of Full Papers Published in Springer CCIS Volume 1342

The papers in this section appear in Springer CCIS Volume 1342
available at https://link.springer.com/book/10.1007/978-3-030-66151-9.
The abstracts are included in this proceedings.

- Mouton, Coenraad; Myburgh, Johannes Christiaan and Davel, Marelie. *Stride and translation invariance in CNNs.*
- Myburgh, Johannes Christiaan; Mouton, Coenraad and Davel, Marelie. *Tracking translation invariance in CNNs.*
- Venter, Arthur; Theunissen, Marthinus and Davel, Marelie. *Pre-interpolation loss behaviour in neural networks.*

Stride and translation invariance in CNNs

Coenraad Mouton[0000−0001−8610−2478], Johannes C. Myburgh[0000−0002−7378−4796], and Marelie H. Davel[0000−0003−3103−5858]

Multilingual Speech Technologies, North-West University, South Africa;
and CAIR, South Africa
{moutoncoenraad, christiaanmyburgh01}@gmail.com
http://engineering.nwu.ac.za/must

Abstract. Convolutional Neural Networks have become the standard for image classification tasks, however, these architectures are not invariant to translations of the input image. This lack of invariance is attributed to the use of stride which subsamples the input, resulting in a loss of information, and fully connected layers which lack spatial reasoning. We show that stride can greatly benefit translation invariance given that it is combined with sufficient similarity between neighbouring pixels, a characteristic which we refer to as *local homogeneity*. We also observe that this characteristic is dataset-specific and dictates the relationship between pooling kernel size and stride required for translation invariance. Furthermore we find that a trade-off exists between generalization and translation invariance in the case of pooling kernel size, as larger kernel sizes lead to better invariance but poorer generalization. Finally we explore the efficacy of other solutions proposed, namely global average pooling, anti-aliasing, and data augmentation, both empirically and through the lens of local homogeneity.

Keywords: Translation invariance · Subsampling · Convolutional Neural Network · Local homogeneity

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_17

Tracking Translation Invariance in CNNs

Johannes C. Myburgh[0000−0002−7378−4796], Coenraad Mouton[0000−0001−8610−2478], and Marelie H. Davel[0000−0003−3103−5858]

[1] Multilingual Speech Technologies, North-West University, South Africa
[2] CAIR, South Africa
{christiaanmyburgh01, moutoncoenraad, marelie.davel}@gmail.com
http://engineering.nwu.ac.za/must

Abstract. Although Convolutional Neural Networks (CNNs) are widely used, their translation invariance (ability to deal with translated inputs) is still subject to some controversy. We explore this question using translation-sensitivity maps to quantify how sensitive a standard CNN is to a translated input. We propose the use of cosine similarity as sensitivity metric over Euclidean distance, and discuss the importance of restricting the dimensionality of either of these metrics when comparing architectures. Our main focus is to investigate the effect of different architectural components of a standard CNN on that network's sensitivity to translation. By varying convolutional kernel sizes and amounts of zero padding, we control the size of the feature maps produced, allowing us to quantify the extent to which these elements influence translation invariance. We also measure translation invariance at different locations within the CNN to determine the extent to which convolutional and fully connected layers, respectively, contribute to the translation invariance of a CNN as a whole. Our analysis indicates that both convolutional kernel size and feature map size have a systematic influence on translation invariance. We also see that convolutional layers contribute less than expected to translation invariance, when not specifically forced to do so.

Keywords: Convolutional Neural Networks · Translation Invariance · Deep Learning.

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_18

Pre-interpolation Loss Behavior in Neural Networks

Arthur E.W. Venter[1,2][0000−0001−7014−8711], Marthinus W. Theunissen[1,2][0000−0002−7456−7769], and Marelie H. Davel[1,2][0000−0003−3103−5858]

[1] Multilingual Speech Technologies (MuST), North-West University, South Africa
[2] CAIR, South Africa
aew.venter@gmail.com, tiantheunissen@gmail.com, marelie.davel@nwu.ac.za

Abstract. When training neural networks as classifiers, it is common to observe an increase in average test loss while still maintaining or improving the overall classification accuracy on the same dataset. In spite of the ubiquity of this phenomenon, it has not been well studied and is often dismissively attributed to an increase in borderline correct classifications. We present an empirical investigation that shows how this phenomenon is actually a result of the differential manner by which test samples are processed. In essence: test loss does not increase overall, but only for a small minority of samples. Large representational capacities allow losses to decrease for the vast majority of test samples at the cost of extreme increases for others. This effect seems to be mainly caused by increased parameter values relating to the correctly processed sample features. Our findings contribute to the practical understanding of a common behavior of deep neural networks. We also discuss the implications of this work for network optimization and generalization.

Keywords: Overfitting · Generalization · Deep learning

This paper is published in the Springer CCIS volume at
https://link.springer.com/chapter/10.1007/978-3-030-66151-9_19

Machine Learning Theory: Full Papers Accepted for SACAIR 2020 Online Proceedings

The following full papers were accepted for inclusion in this proceedings. These papers can be cited as indicated below adding page numbers and the url to the specific paper.

– Burns, Jamie and van Zyl, Terence. *Automated Music Recommendations Using Similarity Learning.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
– Haasbroek, Daniël Gerbrand and Davel, Marelie. *Exploring neural network training dynamics through binary node activations.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
– Lamprecht, Dylan and Barnard, Etienne. *Using a meta-model to compensate for training-evaluation mismatches.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.
– Magangane, Luyolo and Brink, Willie. *Link prediction in knowledge graphs using latent feature modelling and neural tensor factorisation.* In Proceedings of SACAIR 2020. ISBN: 978-0-620-89373-2.

Automated Music Recommendations Using Similarity Learning

Jamie Burns[1] and Terence L van Zyl[2]

[1] School of Computer Science and Applied Mathematics, University of the Witwatersrand, South Africa 1628649@students.wits.ac.za
[2] Institute for Intelligent Systems, University of Johannesburg, South Africa tvanzyl@uj.ac.za

Abstract. Automatic music recommendation is a machine learning problem in which we try to classify music based on the taste of an individual. Due to a continuous shift towards online music store and streaming services, automatic music recommendation remains an increasingly relevant problem. Many modern music recommendation systems make use of collaborative filtering. However, this method fails when there is a lack of historical data available and is ineffective in recommending unfamiliar or unpopular music. In this paper, we explore the use of a latent factor model for music recommendation. We attempt predicting the latent factors that best describe the sound of a song. We compare the use of traditional music features, Mel-frequency cepstral coefficients (MFCCs) with a Siamese recurrent neural network paired with similarity learning. We compare predictions qualitatively on the Free Music Archive (FMA) dataset. We show that not only does the predicted latent factors produce a sensible recommendation but also the use of deep similarity learning is a well-suited method for music recommendation. Our Siamese recurrent neural network significantly outperforms the traditional approach leading us to motivate for further investigation on larger datasets.

1 Introduction

With the increase in the size and usage of digital music collections, the need for more advanced storage mediums has never been more prevalent. An effective music storage medium has other capabilities beyond efficient song retrieval [27]. One such capability is an automated music recommendation.

Automatic music recommendation is a common machine learning problem that classifies music based on the taste of an individual. These systems allow users to discover music, while also assisting online music stores target audiences for their wares. The variety of music styles and genres, as well as social and geographic factors, that can affect a listeners preference, makes music recommendation a challenging task. When considering individual songs, a variety of items can be recommended. This leads to various approaches including album based, and artist based recommendations. Although these approaches are promising and relatively simple to implement, they also limit artist exposure and prevent users from discovering new music. Thus, these approaches may not always be viable solutions to the problem at hand [20].

Music information retrieval (MIR) plays an important role in music recommendatio n[2]. A common approach to MIR is to make use of textual-metadata (which is used to describe the music item of interest). Due to the continuous growth of digital music collections, these have become difficult to maintain. These challenges lead to a proposal for automated systems for MIR [2].

Currently, there are two dominant architectures for MIR, deep and shallow architectures [9]. Shallow architectures transform a time-variant function, a signal into an instantaneous representation, a feature, to extract descriptive features from audio clips. Architectures are defined as shallow if they rely on a single transform to marginalise the temporal dimension of music audio. Deep architectures consist of multiple stacked shallow architectures, each as their own layer. The use of numerous layers allows us to absorb more specific variations in signals, which is difficult to do directly (using shallow architectures) [9]. These deep architectures can be thought of as latent factor models. This is because they can take a high-dimensional feature representation of an audio clip and transform into an unknown lower-dimensional representation, which can be modelled more accurately [23].

Most shallow architecture approaches make use of MFCCs (Mel-frequency Cepstral coefficients), which are short-time spectral decompositions of an audio signal that represents general audio frequency attributes that are imperative to human hearing [16].

Similarity learning is a machine learning technique that is directly related to regression and classification [11]. The goal of similarity learning is, through the use of some similarity function, to learn from examples how best to represent the similarity or dissimilarity between two objects (in our case, songs) [27]. Due to increased data and the need for advanced information retrieval systems, similarity learning has seen increased interest from the scientific community [29] and is a viable technique for MIR.

The subjective nature of music similarity (i.e. opinions on the similarity between songs are not consistent) makes it difficult to collect large amounts of labelled data [2]. The lack of labelled data limits the performance of supervised learning techniques, which are reliant on a large amount of labelled data to make accurate predictions [14]. This provides a potential advantage to unsupervised learning techniques, such as clustering, that do not need the data to be labelled [14]. Further, classification requires a label per class. As a result to transfer to unseen classes a new model will need to be trained. This leads one to consider similarity learning in which embedding can be transferred to unseen classes without requiring retraining [12]. Similarity learning has seen much attention in music and audio information retrieval and has been used in a variety of different audio classification problems. [17] explored the use of similarity learning to rank the similarity between different YouTube audio clips. Although the research focused on environmental sounds, many of the proposed techniques apply to music audio. [13] proposed the use of similarity learning, along with a deep convolutional neural network for genre, mood and instrument classification. Similarly, [21] performed a study on the use of deep architectures in tandem with similarity learning for artist classification. This research on artist classification was further extended by [4], who focused on the use of similarity learning along with shallow architectures for the same task. [10] used similarity for melody retrieval, this has many applications such as matching hummed or sung melodies to songs in a digital music collection, instrument classification and music genre classification. The above studies reveal several benefits to using similarity learning when compared to classical approaches. This leads us to consider similarity learning as a straightforward extension for content-based music recommendation.

In this paper, we explore the use of a deep architecture, along with similarity learning for MIR and recommendation. We compare our results to a shallow approach, that makes use of MFCCs, for automated music recommendation. As such our contributions are:

- We explore the use of similarity learning for music information retrieval and automated music recommendation.
- We extend on existing work by using triplet loss to the train the *Paralleling Recurrent Convolutional Neural Network(PRCNN)* developed by **(author?)** [6], who use *Categorical Cross-Entropy Loss* in their approach.
- We show that similarity learning can be used to automatically extract descriptive audio features, that can be used for accurate music recommendations.

In terms of its impact, the use of similarity learning for automated music recommendation could lead to more accurate music recommendations allowing users to discover music, that better matches their preferences.

2 Methodology

2.1 Dataset

The FMA dataset [5] is a large music archive that consists of 106,574 songs from 16,341 artists and 14,854 albums, arranged in a hierarchical categorisation of 161 genres. The archive contains a variety of different song data, such as audio clips, precomputed audio features (for each song), Echonest features - which are audio features calculated by the *Echonest Analyzer* [24] - and tracks and artist information. The archive can be split into various sizes (small, medium and large) depending on the size of the experiment. We chose to use the *FMA* dataset over a more popular dataset, such as the *Million Song Data (MSD)* [1], due to its inclusion of audio clips.

For our experiments, similar songs were first grouped into classes based on the following labels: artist, album, genre and sub-genre. Songs which had at least two overlapping labels were placed in the same group. After the initial grouping, manual refinement was done to ensure all class labels were correct (i.e. songs within the same groups sounded similar). This grouping technique differs from grouping by genre, as the majority of classes come from the same genres (like Rock, Hip-Hop and Electronic), but are separated based on the album, artist and sub-genre. Music similarity is inherently subjective and as such, each listener has a unique opinion on which songs sound similar and which do not [2]. This subjective nature makes it difficult to label similar sounding music based solely on the album, artist, sub-genre and genre. The manual grouping was as as a result crucial in the dataset creation process. Each audio clip used was 30 seconds long and we used 80% of the data for training, 10% for validation and 10% for testing.

2.2 Audio features

For the baseline approach, the *Mel-frequency cepstral coefficients (MFCC)* for each song were used. These are calculated by taking the logarithm of the Mel magnitude spectrum and decorrelating the resulting values using a Discrete Cosine Transform [2]. These features were calculated using a window length of 2048 and a hop size of 512. The mean value for the MFCCs was extracted at each frame which resulted in a 1-dimensional 140 features vector for each song. These features were utilised directly to measure the similarity between songs.

For our approach, we used the *Log-scaled Mel-spectograms*. This is achieved by log scaling the Mel-spectrograms from each song. This created a feature vector of shape (640x128). Similarly to the baseline, these feature values were calculated using a window length of 2048 and a hop size of 512 [5]. The Mel-spectrograms were then sent through a deep architecture to extract the most meaningful features for each song.

2.3 Network Architecture

Inspired by the work of [6], a *Paralleling Recurrent Convolutional Neural Network(PRCNN)* was used in our implementation. This specific architecture was preferred due to its ability to preserve the temporal relationships of the original signals in the music, which is not possible using a standard Convolutional Neural Network (CNN). In our implementation we use *triplet loss* to train the *PRCNN*, and utilize the network for feature extraction rather than classification. The network consists of two blocks, the *Convolutional (CNN) block* and the *Bidirectional Gated Recurrent Units Block (BGRU-RNN)*, as seen in Figure 1. The feature vector, described in Section 2.2, is fed through both blocks in parallel.

The CNN block consists of 5 convolutional layers, with 16, 32, 64, 128 and 64 filters respectively. Between each convolutional layer, there are max-pooling layers. The first three max-pooling layers have a pool size of (2x2), and the upper two max-pooling have a pool size of (4x4). A *ReLu activation function* [7] was used for all of the convolutional layers and is described as:

$$R(z_i) = \begin{cases} z_i & z_i > 0 \\ 0 & z_i \leq 0 \end{cases},$$

where z_i is the input value at a given node i. The output from the convolutional block is a 256-dimensional vector.

For the BGRU-RNN block, the first layer is a max-pooling layer with a pool size of (4x2). Following the max-pooling layer is an embedding layer, for further dimensional reduction. The reduced vector is then sent to the bidirectional GRU with 64 units. The output from the BGRU-RNN block is a 128-dimensional vector.

The output from both blocks is then concatenated into a 384-dimensional vector. Finally, the concatenated vector is passed through a dense layer, which uses a *Sigmoid activation function* [19]:

$$S(x) = \frac{1}{1 + e^{-x}}.$$

The dense layer returns a 60-dimensional feature vector. The overall architecture is illustrated in Figure 1.

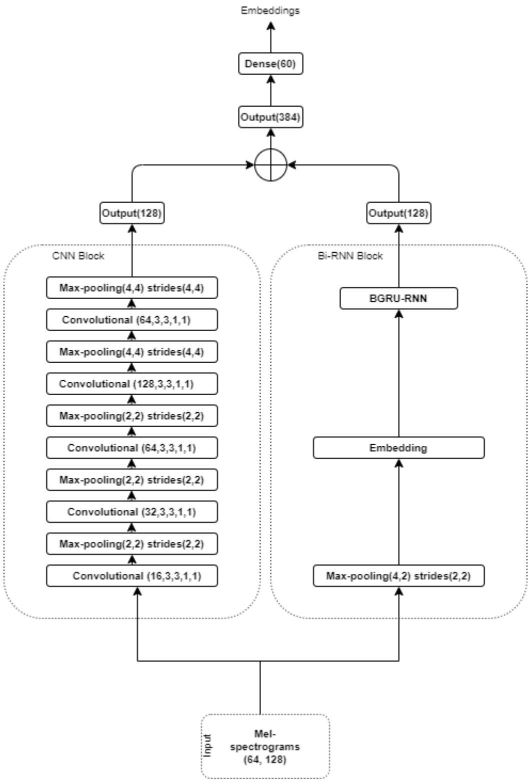

Embeddings
Dense(60)
Output(384)
Output(128)
Output(128)
CNN Block
Max-pooling(4,4) strides(4,4)
Convolutional (64,3,3,1,1)
Max-pooling(4,4) strides(4,4)
Convolutional (128,3,3,1,1)
Max-pooling(2,2) strides(2,2)
Convolutional (64,3,3,1,1)
Max-pooling(2,2) strides(2,2)
Convolutional (32,3,3,1,1)
Max-pooling(2,2) strides(2,2)
Convolutional (16,3,3,1,1)
Bi-RNN Block
BGRU-RNN
Embedding
Max-pooling(4,2) strides(2,2)
Input
Mel-
spectrograms
(64, 128)

Fig. 1: The network architecture of the PRCNN

The network architecture in Figure 1 was employed as a *Siamese Neural Network* to be trained using *Triplet Loss.* This involves creating 3 identical versions of the same *PRCNN* and updating the weights of all 3 networks with the same values dat each step of the training process (ie. throughout the entire training process the networks and their corresponding weights remain identical). For the prediction of the latent factors (following the training phase), we use only one of the three networks (because the networks are identical the one we choose does not matter).

2.4 Loss Function

The *PRCNN* was trained using two different loss functions, namely, *Triplet Loss* and *Categorical Cross Entropy Loss. Triplet loss* is described below as [25]:

$$\text{basic loss} = \Sigma_{i=1}^{N}\left[||f_i^a - f_i^p||^2 - ||f_i^a - f_i^n||^2 + \alpha\right]$$

where f_i^a, f_i^p and f_i^n, respectively represent the latent factor vectors of the anchor, positive and negative samples and $\alpha > 0$ is the margin of similarity (which can be increased to further separate embeddings).

This loss ensures that a given anchor point x_a and a positive point x_p belonging to same class y_a, are closer together than the same anchor point and a negative point x_n belonging to a different class y_n, by at least a margin α [30]. Furthermore any triple sample (anchor, positive, negative) that generated a negative *basic_loss* was set to 0 [25]:

$$\text{loss} = \max\{basic\_loss, 0.0\}$$

To optimise the performance of the triplet loss function a variety of mining strategies were employed, *Batch Random, Batch Semi-Hard* and our own batch strategy, the *custom batch strategy.* The *Batch Random* strategy assigned a random anchor with a random positive sample and a random negative sample. This strategy was used as a baseline to compare the other batch strategies to. The *Batch Semi-Hard* [30] strategy assigned positive and negative samples to a given anchor such that the negative sample was further away from the anchor than the positive sample, but closer, to the anchor, than the positive sample plus the margin α given by:

$$\Sigma_{i=1}^{N}||f_i^a - f_i^p||^2 < \Sigma_{i=1}^{N}||f_i^a - f_i^n||^2 < \Sigma_{i=1}^{N}||f_i^a - f_i^p||^2 + \alpha$$

Finally our batch strategy, the *custom batch* strategy, pairs each anchor with all positive samples and assigns only the hardest negative sample, that is the closest negative sample to the anchor, to each of the anchor, positive pairs to create the triplets.

Before utilizing any of the batch strategies mentioned above, the *PRCNN* was pre-trained for jump start and transfer learning purposes using *Categorical Cross Entropy Loss* [8]:

$$\text{loss} = -\Sigma_{b=1}^{B}\Sigma_{j=1}^{M}(y_j^{(b)}\log(\sigma(z_i^{(b)}))$$

Where M is the number of classes, B is the batch size and y is the binary indicator (0 or 1) for if class label j is the correct class label (as described in Section 2.1) for observation b. Finally $\sigma(z_i)$ is the output of the final layer, which is a *Softmax* activation function and is described as follows:

$$\sigma(z_i) = \frac{e^{z_i}}{\Sigma_{j=1}^{M} e^{z_j}}.$$

Note during this pre-training phase an extra *dense* layer was added to the *PRCNN* (which was later dropped when changing to *Triplet Loss*). This layer was made of 8 units and used a *Softmax* activation function (described in Equation 2.4). This final layer allowed us to use *Categorical Cross-entropy* to pre-train the network.

As a separate experiment, we also trained the *PRCNN* (to convergence) using *Categorical Cross-Entropy Loss* (Described in Equation 2.4). This version of the *PRCNN* was exactly the same as the version used to pre-train the *PRCNN* (for *Triplet Loss*). The final dense layer of the network was popped off, when predicting latent factors, to ensure the dimensions of the latent factor vectors remained consistent for all experiments. This final experiment was used as another baseline to compare the results of the *PRCNN* trained using *Triplet Loss* too.

2.5 Experiments

To test the effectiveness of the described latent factor models, i.e. the output from the *PRCNN* (using all batch strategies) as well as the version trained using *Categorical Cross-Entropy Loss*, and the baseline feature vector (made up of the MFCC features described in Section 2.2), a *KNN algorithm* along with *Cosine similarity* was used to group both the similar latent factors and the similar *MFCC* features. This algorithm was run on all feature/latent factor sets independently.

Given a song, the aim is to recommend N of the most similar sounding songs, by measuring the similarity between each songs respective feature embedding. To measure the similarity between two feature vectors, Cosine similarity [18] was used:

$$\text{similarity} = \frac{A \cdot B}{||A||||B||} = \frac{\Sigma_{i=1}^{n} A_i B_i}{\sqrt{\Sigma_{i=1}^{n} A_i^2}\sqrt{\Sigma_{i=1}^{n} B_i^2}}$$

where A and B represent the feature vectors of two different song's. Cosine similarity was chosen due to its benefits over other metrics such as *Euclidean Distance* and *Manhattan Distance*, for high dimensional vectors, where the magnitude, of the vector, is not important.

2.6 Hyper-parameters

To optimise the hyperparameters of the *PRCNN*, 8-fold cross-validation was used. The technique involves separating a shuffled set of input data and dividing it into 8 equal-sized sets. For each respective run, a different set is used for validation and testing. The final 6 sets for each run are used to perform training on the model. The testing accuracy from the model is averaged across all 6 runs to determine which hyperparameters provided the overall best results. *Early stoppage* was also used to prevent the network from overfitting. Finally, the optimiser function used was *Stochastic Gradient Descent* [22]. We kept the learning rate consistent for all experiments at 0.001.

2.7 Metrics

To measure the performance of our music recommendation system the following metrics are utilised:

Mean Reciprocal Rank (MRR) *Mean Reciprocal Rank (MRR)* is an indication of how high the first correct song recommendation is overall queries, and is described below as:

$$MRR = \frac{1}{|Q|} \Sigma_{i=1}^{|Q|} \frac{1}{rank_i}$$

where $||Q||$ and $rank_i$ respectively denote the number of queries and the position of the first relevant result.

Mean Average Precision (MAP) *Mean Average Precision (MAP)* is a measure of the average recommendation precision over all queries, and is described below as:

$$MAP = \frac{1}{||Q||} \Sigma_{q=1}^{||Q||} AP(q)$$

where $AP(q)$ denotes the average precision of a given query, q.

Precision@k (P@k) *Precision@k (P@K)* indicates how many of the first k recommendations came from the same class as the query, and is described as:

$$Precision@k = \frac{\text{true positives@}k}{(\text{true positives@}k\ ) + (\text{false positives @}k)}$$

Where k has been set to 1, 5 and 9 respectively.

3 Results

The overall *MAP* and *MRR* as well as the *Precision@1*, *Precision@5* and the *Precision@9* are all shown in Table 1. Note the results in Table 1 and Table 2 were retrieved by averaging the results after running each of the experiments, explained in Section 2.5, 30 times.

Table 1: The mean MAP, MRR, Precision@1, Precision@5 and Precision@9 score for each respective feature set. Note that the highest score for each metric is in bold

Features	**MRR**	**MAP**	**P@1**	**P@5**	**P@9**
Baseline Features (MFCC)	0.780 ± 0.0006	0.664 ± 0.0010	0.623 ± 0.0028	0.0.515 ± 0.0008	0.455 ± 0.0003
Random Batch Strategy	0.816 ± 0.0015	0.724 ± 0.0012	0.679 ± 0.0047	0.630 ± 0.0022	0.580 ± 0.0018
Semi-hard Batch Strategy	**0.900 ± 0.0003**	**0.845 ± 0.0002**	**0.827 ± 0.0013**	**0.750 ± 0.0017**	**0.687 ±0.0017**
Custom Batch Strategy	0.856 ± 0.0016	0.773 0.0014	0.750 ± 0.0041	0.694 ± 0.0022	0.637 ± 0.0020
Categorical Crossentropy Loss	0.862 ± 0.0014	0.785 ± 0.0014	0.749 ± 0.0035	0.0.721 ± 0.002	0.658 ± 0.0012

The *TSNE* algorithm [15] was used to plot the various feature emebddings in a 2D plane, shown in Figures 2, 3, 4, 5 and 6. Finally, Table 2 shows the *MAP* and *MRR* score for each class individually. Each column refers to one of the feature sets described in Section 2.5, while each row represent each "class" of similar sounding music.

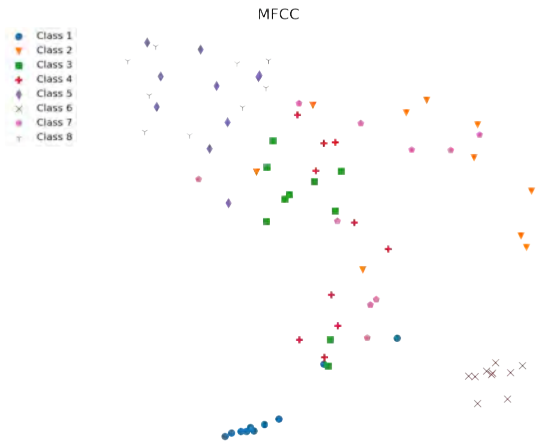


Fig. 2: *TSNE* Plot for *MFCC* features

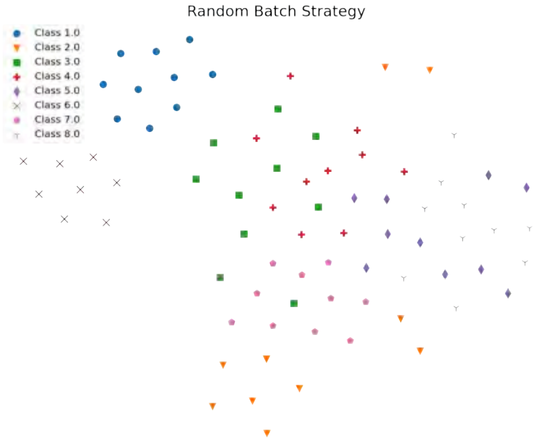


Fig. 3: *TSNE* Plot for *Random batch* features

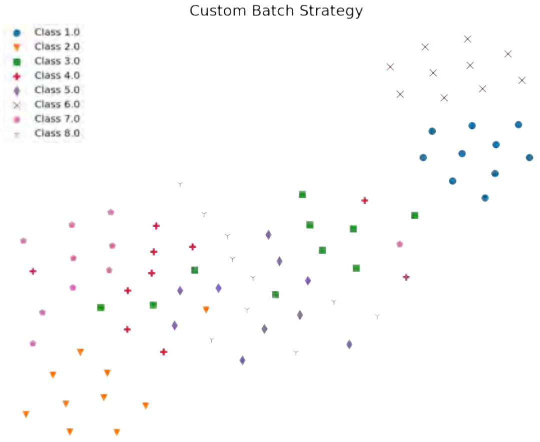


Fig. 4: *TSNE* Plot for *Custom batch* features

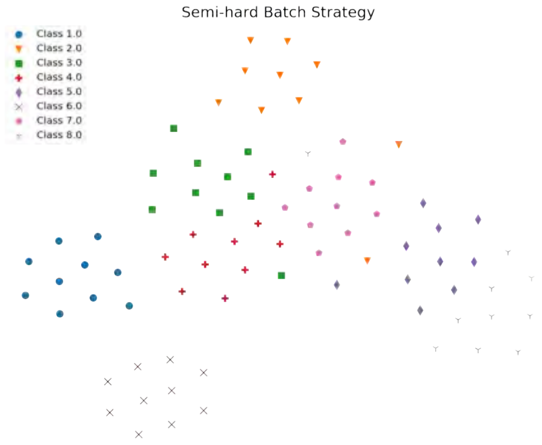


Fig. 5: *TSNE* Plot for *Semi-hard batch* features

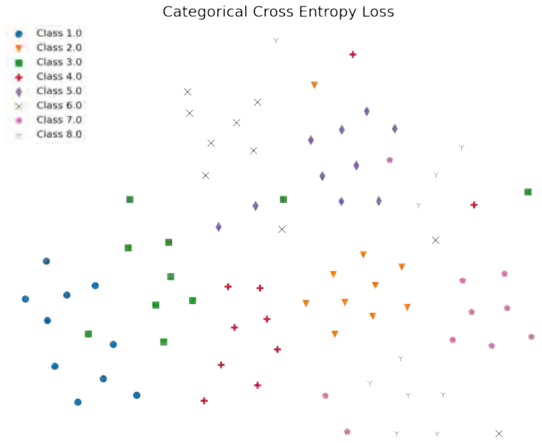


Fig. 6: *TSNE* Plot for *Categorical Crossentropy Loss* features

Table 2: The mean MAP and MRR score of each feature set for each unique music class. Note the highest average of the MAP and MRR scores, for each class, is highlighted in bold (CLL is an abbreviation for Categorical Cross Entropy Loss).

Strategy	**Custom Batch**		**Random Batch**		**Semi-hard Batch**		**Baseline(MFCC)**		**CCL**	
Class	**MAP**	**MRR**	**MAP**	**MRR**	**MAP**	**MRR**	**MAP**	**MRR**	**MAP**	**MRR**
Class 1	**1.0 ± 0.0**	**1.0 ± 0.0**	0.99 ± 0.0	0.99 ± 0.0	**1.0 ± 0.0**	**1.0 ± 0.0**	0.98 ± 0.0	0.99 ± 0.001	**1.0 ± 0.0**	**1.0 ± 0.0**
Class 2	0.68 ± 0.01	0.80 ± 0.009	0.67 ± 0.026	0.77 ± 0.031	**0.83 ± 0.006**	**0.96 ± 0.003**	0.62 ± 0.008	0.75 ± 0.013	0.73 ± 0.009	0.81 ± 0.01
Class 3	0.85 ± 0.007	0.92 ± 0.003	0.70 ± 0.010	0.80 ± 0.010	**0.89 ± 0.003**	**0.93 ± 0.003**	0.57 ± 0.011	0.70 ± 0.007	0.82 ± 0.008	0.9 ± 0.004
Class 4	0.54 ± 0.017	0.64 ± 0.017	0.56 ± 0.007	0.73 ± 0.003	0.59 ± 0.004	0.72 ± 0.010	0.59 ± 0.010	0.69 ± 0.024	**0.62 ± 0.024**	**0.78 ± 0.017**
Class 5	0.83 ± 0.009	0.93 ± 0.009	0.64 ± 0.006	0.78 ± 0.007	**0.93 ± 0.004**	**0.97 ± 0.001**	0.41 ± 0.008	0.64 ± 0.014	0.77 ± 0.15	0.87 ± 0.011
Class 6	0.092 ± 0.0	1.0 ± 0.0	0.97 ± 0.001	0.99 ± 0.0	**1.0 ± 0.0**	**1.0 ± 0.0**	0.93 ± 0.001	0.99 ± 0.0	**1.0 ± 0.0**	**1.0 ± 0.0**
Class 7	0.63 ± 0.010	0.78 ± 0.012	0.58 ± 0.007	0.68 ± 0.018	**0.74 ± 0.007**	**0.86 ± 0.005**	0.53 ± 0.012	0.67 ± 0.01	0.65 ± 0.006	0.75 ± 0.007
Class 8	0.66 ± 0.008	0.77 ± 0.014	0.68 ± 0.009	0.78 ± 0.017	0.62 ± 0.11	0.76 ± 0.020	0.68 ± 0.006	0.8 ± 0.005	**0.69 ± 0.008**	**0.79 ± 0.008**

4 Discussion

4.1 Evaluation

From Table 1 we note that all latent factors extracted using the *PRCNN*, regardless of the chosen batch or training strategy, scored a higher *MAP* and *MRR* score than the MFCC baseline features. The latent factors extracted using the version of the *PRCNN* trained using *catergorical cross-entropy loss* outperformed the latent factors extracted using the *random batch* strategy, while also scoring very similar results (with even a slight advantage) to the latent factors extracted using the *custom batch* strategy. The highest achieved *MRR* and *MAP* scores were 0.900 and 0.845 respectively, and both were achieved using the *Semi-hard Batch Strategy*.

Performance within the different batch strategies was as expected, with the *Random Batch Strategy* performing the worst, followed by the *Custom Batch Strategy* and finally, the best performing tested batch strategy was the *Semi-hard Batch Strategy*. The *Random Batch Strategy* is naive and so was unable to make clear distinctions between classes which shared some degree of similarity. The *Custom Batch Strategy* is slightly more intelligent when creating triplets, and so we see slight improvements over the *Random Batch Strategy*. Finally, significant improvements were seen, over the other two batch strategies, when using the *Semi-hard Batch Strategy*, this can be attributed to the selection of triplets that lead to the greatest degree of separability between different classes of music. It is important to note that we also tested a *Hard Batch Strategy*[30], but using this strategy, the *BGRU-RNN* was unable to converge to a local minimum, and thus almost no distinctions were made between classes.

From Figures 2, 3, 4, 5 and 6 we note that some classes were more separable than others. Certain classes, such as class 1 and class 6, performed well for all features sets, while other classes, such as class 4, performed poorly throughout. Some overlap between classes is to be expected, as certain music features are consistent across all classes, and these features are almost impossible to neglect when measuring the similarity between songs. The results observed in figures 2, 3, 4 and 5 are further supported by Table 2, where we see high *MRR* and *MAP* scores for classes 1 and 6 across all feature sets, while class 4 comparatively performed poorly.

4.2 Related Work

Similar sounding music is, for the most part, subjective, and as such, it is difficult to make a direct comparison between our results, and what is seen in other relevant literature. Although there are certain consistencies within approaches which are important to note.

[6] used a *BGRU-RNN* for *Music Genre Classification* on the GTZAN [28] dataset. Each 30 second audio clip was converted to a Short-time Fourier Transform (STFT) [26] with dimensions 128x513. The network was trained using *Categorical Cross-Entropy Loss* and classification was done in the final layer rather than the feature embedding learning in our approach. This work differs further from our approach, where the *BGRU-RNN* was used rather as a latent factor predictor and *Triplet Loss* was the selected loss function.

[20] proposed the use of content-based music recommendation as a potential substitute for more common *collaborative filtering* methods. They explored 2 approaches to content-based music recommendation, namely, a *Bag-of-Words* representation of *MFCCs* paired with linear regressions and a *Convolutional Neural Network (CNN)*, and compared them with *collaborative filtering methods* from relevant literature. Tests were run on *Million Song Dataset (MSD)* [1], which is significantly larger than our dataset, and so it's hard to make a direct comparison between results. Although, it is important to note that there are certain consistencies, between our results and theirs, with their deep architecture, a non-recurrent *CNN*, outperforming the shallow architecture, *MFCCs*, which is similar to what we saw in our results.

Finally, [3] conducted a comparative study of the benefits of a *Convolutional Recurrent Neural Network (CRNN)* over a *Convolutional Neural Network* for music genre classification. Similarly to [20], they ran tests on the *Million Song Dataset* [1] and so again it is hard to make comparisons between their results and ours. They found that both network architectures have potential benefits for music genre classification. Our chosen network architecture, the *BGRU-RNN*, consists of both a *CNN* component and a *RNN* but connected in parallel, which is in contrast to the *CRNN* in their implementation, where they have them connected in series.

5 Conclusions

In this paper, we proposed the use of similarity learning as a feature learning architecture for music audio. We compared audio features extracted using similarity learning to baseline features, MFCCs, and tested the capabilities of both feature sets in the context of content-based music recommendation. Our results showed that not only is similarity learning a viable approach to audio feature extraction, but, in the context of automated music recommendation, showed significant improvements over the baseline features, the MFCCs and even outperformed the same network trained with a different, more common loss function, namely *categorical cross-entropy loss*. Furthermore, from the tested batch strategies, we found the *Semi-hard batch strategy*, was the most effective for audio feature extraction. Finally, we observed that some classes of music are more differentiable than others, and as such performed better for tested feature extraction methods. In the future, we would like sample features from the entire song rather than just a 30-second clip. We would also like to test various other similarity loss functions, such as angular and pairwise loss.

References

[1] Bertin-Mahieux, T., Ellis, D.P., Whitman, B., Lamere, P.: The million song dataset (2011)

[2] Casey, M.A., Veltkamp, R., Goto, M., Leman, M., Rhodes, C., Slaney, M.: Content-based music information retrieval: Current directions and future challenges. Proceedings of the IEEE **96**(4), 668–696 (2008)

[3] Choi, K., Fazekas, G., Sandler, M., Cho, K.: Convolutional recurrent neural networks for music classification. In: 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 2392–2396. IEEE (2017)

[4] Cleveland, J., Cheng, D., Zhou, M., Joachims, T., Turnbull, D.: Content-based music similarity with triplet networks. arXiv preprint arXiv:2008.04938 (2020)

[5] Defferrard, M., Benzi, K., Vandergheynst, P., Bresson, X.: Fma: A dataset for music analysis. arXiv preprint arXiv:1612.01840 (2016)

[6] Feng, L., Liu, S., Yao, J.: Music genre classification with paralleling recurrent convolutional neural network. arXiv preprint arXiv:1712.08370 (2017)

[7] Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier neural networks. In: Proceedings of the fourteenth international conference on artificial intelligence and statistics. pp. 315–323 (2011)

[8] Gómez, R.: Understanding categorical cross-entropy loss, binary cross-entropy loss, softmax loss, logistic loss, focal loss and all those confusing names. URL: https://gombru. github. io/2018/05/23/cross_ entropy_loss/(visited on 29/03/2019) (2018)

[9] Humphrey, E.J., Bello, J.P., LeCun, Y.: Moving beyond feature design: Deep architectures and automatic feature learning in music informatics. In: ISMIR. pp. 403–408. Citeseer (2012)

[10] Karsdorp, F., van Kranenburg, P., Manjavacas, E.: Learning similarity metrics for melody retrieval. In: ISMIR. pp. 478–485 (2019)

[11] Kaya, M., Bilge, H.Ş.: Deep metric learning: A survey. Symmetry **11**(9), 1066 (2019)

[12] Kulis, B., et al.: Metric learning: A survey. Foundations and trends in machine learning **5**(4), 287–364 (2012)

[13] Lee, J., Bryan, N.J., Salamon, J., Jin, Z., Nam, J.: Disentangled multidimensional metric learning for music similarity. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 6–10. IEEE (2020)

[14] Lison, P.: An introduction to machine learning. Language Technology Group (LTG) **1**(35) (2015)

[15] Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(Nov), 2579–2605 (2008)

[16] Mandel, M.I., Ellis, D.P.: Song-level features and support vector machines for music classification (2005)

[17] Manocha, P., Badlani, R., Kumar, A., Shah, A., Elizalde, B., Raj, B.: Content-based representations of audio using siamese neural networks. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 3136–3140. IEEE (2018)

[18] Nguyen, H.V., Bai, L.: Cosine similarity metric learning for face verification. In: Asian conference on computer vision. pp. 709–720. Springer (2010)

[19] Njikam, A.N.S., Zhao, H.: A novel activation function for multilayer feed-forward neural networks. Applied Intelligence **45**(1), 75–82 (2016)

[20] Van den Oord, A., Dieleman, S., Schrauwen, B.: Deep content-based music recommendation. In: Advances in neural information processing systems. pp. 2643–2651 (2013)

[21] Park, J., Lee, J., Park, J., Ha, J.W., Nam, J.: Representation learning of music using artist labels. arXiv preprint arXiv:1710.06648 (2017)

[22] Robbins, H., Monro, S.: A stochastic approximation method. The annals of mathematical statistics pp. 400–407 (1951)

[23] Sammut, C., Webb, G.: Latent factor models and matrix factorizations (2010)

[24] Schindler, A., Rauber, A.: Capturing the temporal domain in echonest features for improved classification effectiveness. In: International Workshop on Adaptive Multimedia Retrieval. pp. 214–227. Springer (2012)

[25] Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 815–823 (2015)

[26] Sigtia, S., Dixon, S.: Improved music feature learning with deep neural networks. In: 2014 IEEE international conference on acoustics, speech and signal processing (ICASSP). pp. 6959–6963. IEEE (2014)

[27] Slaney, M., Weinberger, K., White, W.: Learning a metric for music similarity. In: International Symposium on Music Information Retrieval (ISMIR). vol. 148 (2008)

[28] Tzanetakis, G., Cook, P.: Gtzan genre collection. Music Analysis, Retrieval and Synthesis for Audio Signals (2002)

[29] Wang, J., Zhang, T., Sebe, N., Shen, H.T., et al.: A survey on learning to hash. IEEE transactions on pattern analysis and machine intelligence **40**(4), 769–790 (2017)

[30] Xuan, H., Stylianou, A., Pless, R.: Improved embeddings with easy positive triplet mining. In: The IEEE Winter Conference on Applications of Computer Vision. pp. 2474–2482 (2020)

Exploring neural network training dynamics through binary node activations

Daniël G. Haasbroek[0000−0002−9974−3626] and
Marelie H. Davel[0000−0003−3103−5858]

Multilingual Speech Technologies (MuST), North-West University, South Africa; and
CAIR, South Africa.

Abstract. Each node in a neural network is trained to activate for a specific region in the input domain. Any training samples that fall within this domain are therefore implicitly clustered together. Recent work has highlighted the importance of these clusters during the training process but has not yet investigated their evolution during training. Towards this goal, we train several ReLU-activated MLPs on a simple classification task (MNIST) and show that a consistent training process emerges: (1) sample clusters initially increase in size and then decrease as training progresses, (2) the size of sample clusters in the first layer decreases more rapidly than in deeper layers, (3) binary node activations, especially of nodes in deeper layers, become more sensitive to class membership as training progresses, (4) individual nodes remain poor predictors of class membership, even if accurate when applied as a group. We report on the detail of these findings and interpret them from the perspective of a high-dimensional clustering process.

Keywords: Neural networks · Generalisation · Clustering

1 Introduction

Deep neural networks (DNNs) have been used to solve increasingly difficult tasks with increasingly high accuracy, and are particularly successful when modelling complex relationships from large quantities of high-dimensional data [8]. While DNN models perform extremely well given sufficient training data, the DNN training process itself is computationally inefficient, with model optimisation requiring expensive searches across a large number of interacting hyperparameters. This search process is mainly guided by heuristics, and by tracking performance on training and held-out validation sets, as no comprehensive theoretical framework yet exists with which to reason about the training process or the expected ability of the optimised models to generalise to out-of-sample data.

The generalisation ability of DNNs has been the topic of much controversy and has been studied from a variety of perspectives. Studies that aim to characterise and predict the generalisation ability of DNNs include approaches that consider the complexity of the hypothesis space, the geometry of the loss surface, characteristics of the classification margins, and statistical measures of uniform stability and robustness. (See Section 2.1.) While each approach provides additional insight, a general analysis framework of DNN behaviour remains elusive. The 'apparent paradox' that DNNs are able to generalise well despite extremely large capacity remains largely unresolved [14].

With 'DNN behaviour' we refer to the performance of a DNN during and after training, as measured on different subsets of the data, both seen during training and not. Characterising this behaviour should allow us to reason about the training process and differences among networks, and to predict characteristics that lead to better performance.

One approach towards probing DNN behaviour is to consider nodes as individual classifiers, collaborating in solving a network-wide task [4]. A consequence of this analysis is that each individual node is implicitly associated with the specific cluster of samples for which it activates. These sample clusters then become useful elements in analysing network behaviour. Specifically, as any node delineates a region in input space for which it activates, any samples that fall within this region are in effect clustered together. These clusters are then used to refine the weights linked to the specific node, improving cluster boundaries.

While the potential importance of these clusters during the training process has been highlighted [4,27], their evolution during training has not yet been explored. Towards this goal, we train several ReLU-activated MLPs on a simple classification task (MNIST) and track the process whereby these sample sets are formed during training. The **main contributions** of this paper are the following:

1. We provide additional motivation for the potential importance of 'sample sets' (the set of samples that activates an individual node) and their corresponding sample-feature clusters when analysing DNN behaviour.
2. We report on the evolution of these clusters during the training process of different fully-connected feedforward networks, and demonstrate that a consistent training process emerges.
3. We interpret these findings in terms of a high-dimensional clustering process, which we conjecture to be a useful perspective when analysing the generalisation ability of neural networks.

We first present relevant background (Section 2), before discussing our motivation for studying sample clusters (Section 3). A description of the analysis approach follows in Section 4, with Section 5 measuring and reporting on the the process whereby sample sets evolve during training. Findings are discussed in Section 6, and summarised in Section 7.

2 Background

We briefly discuss the generalisation ability of DNNs and approaches towards studying this. We then review earlier work related to the role and analysis of sample sets, specifically focusing on sample sets as an element in probing the generalisation ability of DNNs.

2.1 Generalisation in DNNs

DNNs are well understood from the perspective of expressivity: it is known that even a shallow network with sufficient nodes and non-linear activations is able to approximate any function, given some caveats that typically do not apply to real-world data [24]. Similarly, gradient-based optimisation procedures are theoretically well-grounded, with the conditions for finding minima known. Being expressive and trainable, however, does not imply the ability to generalise well; this requires a model to accurately capture the 'true' underlying data distribution, identifying relevant features and their interaction throughout the input domain.

Statistical learning theory (SLT) suggests that the generalisation error of a trained model is bounded above by the complexity of the hypothesis space of the modelling method [2,11]. This bound explains the generalisation ability of many model types, in the sense that an increase in the complexity of the hypothesis space, beyond the amount necessary to capture important relationships, leads to poorer generalisation of trained models [11]. When the complexity of the hypothesis space is higher than needed, good generalisation can still be obtained by introducing a preference for certain functions in the hypothesis space [10]. The use of such regularisation techniques also explains the generalisation ability of many model types [10,28].

For DNNs, neither the complexity of the hypothesis space of a particular network nor the use of regularisation techniques during training can adequately explain generalisation [28]. This was demonstrated in [28] and [26], among others, where the authors obtained good generalisation with networks that could, without any modification, fit random data easily, regardless of the use of regularisation techniques.

A significant body of work has studied generalisation in DNNs. In addition to a host of empirical studies [1,17,22,23], we highlight four approaches:

- **Geometry of the loss landscape.** It has been argued that the smoothness of the loss landscape and, specifically, the flatness of minima can lead to better generalisation [12,15]. This follows the intuition that, under these conditions, an applicable minimum should be more easily accessible during gradient descent, and small perturbations in either input or parameter space should not influence model behaviour. In high-dimensional space, however, it is extremely difficult to obtain a consistent perspective of the error surface, and it has been shown that this error surface can fairly easily be manipulated with little effect on generalisation [5].
- **Statistical measures.** Both the stability of the training process (stability when trained on different datasets) and its robustness (expected behaviour when trained on all possible datasets) have produced insights into the generalisation behaviour of DNNs [9,25,18]. Kawaguchi et al. [14] argue convincingly that there is no paradox when applying such measures from SLT to DNNs; rather, their direct applicability is restricted.
- **Complexity of the hypothesis space.** Complexity can be reduced through regularisation (as discussed above) or through inducing sparsity. With a

smaller set of trainable parameters, prediction accuracy is expected to be more stable, as justified by SLT [21]. To date, sparsity measures have been more useful in improving the computational cost and interpretability of networks than in predicting generalisation ability [7,19].

- **Margin distributions.** These approaches study the decision margin in order to explain generalisation ability, as has been successfully done with linear models such as support vector machines (SVMs). Results related to DNNs [6,13] are promising, with [13], specifically, demonstrating that a linear model, trained on the margin distributions of numerous DNNs, can predict generalisation error.

These approaches tend to consider the network as a whole, with less attention paid to the role of the individual subcomponents — an approach taken in [4], and discussed in more detail below.

2.2 Sample sets

Simultaneously introduced in [3,4,26], the term 'sample set' refers to the node-specific set of samples that activate a given node. The initial definition arose in the context of a *ReLU-activated*, fully-connected feedforward network applied to a classification task; this is also the context we use for the current discussion.

In [4], a DNN is viewed as consisting of layers of local classifiers, collaborating to solve the overall classification task. During gradient descent, weights linked to a node are only updated based on those samples that activate the node. Gradient descent can then be viewed as a two-step process: (1) during the forward pass, sample sets are created; (2) during the backward pass and parameter-update step, the sample sets are refined [4]. This refinement process is both locally and globally aware: at the local level, only selected samples (those in the sample set) are utilised to update local parameters; globally, the loss value associated with each sample takes all network parameters into account [4].

In a related study [26], extended in [27], sample-set composition is analysed in the presence of different types of noise. Interestingly, if the features of training samples are corrupted (in contrast with corrupting labels) high levels of noise — up to 90% for the tasks studied — can be absorbed by the models, without causing any detrimental effect on classification performance. Probing this behaviour revealed that nodes tend to have sample sets that contain either true or corrupted samples, rather than both. Features attuned to the noisy samples, therefore, have almost no effect on noiseless test results. This provides a simple mechanism for preventing additional parameters from hurting performance: additional parameters create an increased number of sub-components with which to isolate detrimental samples from the rest of the training data, without fundamentally changing the way in which the uncorrupted samples are modelled [27].

In both [26] and [3] it is noted that the sample set of any node is fully described by that node's fan-in weight vector; if the activation vector in the prior layer is aligned with this weight vector (if their dot product is positive), the node will activate, otherwise not. The weight vector, therefore, creates a boundary that separates samples included in the current sample set from those excluded. Finally, in [4] it is shown that, in deeper layers, sample sets become class-sensitive, containing either almost all or almost none of the samples of any class. This behaviour is consistent across a range of architectures [4].

In summary, the above findings indicate that the training process creates clusters of samples with very specific characteristics. This points towards sample sets as a useful element in understanding network behaviour, a view that we explore further in Section 3.

3 Motivation for exploring sample-feature clusters

As discussed in Section 2.2, sample sets identify regions in the input space where specific features produce coherent behaviour. This can also be understood by viewing the process whereby samples are included or excluded from these sets. As in [4], we focus on ReLU-activated feedforward networks and define[1] the sample set $\hat{S}_{b,l,j}$ at node j of layer l as those samples in batch b that produce a positive activation value at node j. For any sample in the sample cluster, the node j in layer l can be connected to an arbitrary number of active nodes in layer $l+1$. By selecting one active node per layer, the weights connecting these active nodes can be used to define an active path $p = \{w_{p_1}, w_{p_2}, \ldots, w_{p_{N-l}}\}$ associated with a specific sample, starting at layer l and ending at a node in the output layer N.

During standard SGD training, the sample set (and only the sample set) influences the weight update. If we initially limit our analysis to networks with no bias values beyond the first layer, the weight update equation simplifies considerably. As derived in [4], the SGD weight update $\delta w_{l,j,i}$ for the weight connecting node i to node j at layer l is then given by

$$\delta w_{l,j,i} = -\eta \sum_{\mathbf{s} \in \hat{S}_{b,l,j}} \left[z^{\mathbf{s}}_{l-1,i} \sum_{p \in P^{\mathbf{s}}_j} \left(\lambda^{\mathbf{s}}_p \prod_{k=1}^{N-l} w_{p_k} \right) \right] , \tag{1}$$

where the superscript $\mathbf{s}$ indicates sample-specific values, η indicates the learning rate, $z^{\mathbf{s}}_{l-1,i}$ indicates the post-activation value at node i in layer $l-1$, $P^{\mathbf{s}}_j$ indicates the set of all active paths linking node j to the output layer, and $\lambda^{\mathbf{s}}_p$ indicates the derivative of the loss function with respect to the network output.

This sample-specific weight update can be expressed in terms of the *node-supported cost*, a scalar value that represents the portion of the final cost that can be attributed to all active paths emanating from node j, when processing sample $\mathbf{s}$ [3]. Specifically, the sample-specific node-supported cost at layer l, node j can be defined as

$$\phi^{\mathbf{s}}_{l,j} = \sum_{p \in P^{\mathbf{s}}_j} \left(\lambda^{\mathbf{s}}_p \prod_{k=1}^{N-l} w_{p_k} \right) . \tag{2}$$

[1] Note that we follow the derivations from [4] but use a different notation, for better clarity.

The update to the weight vector $\mathbf{w}_{l,j}$ feeding into node j at layer l is then given by

$$\delta\mathbf{w}_{l,j} = -\eta \sum_{\mathbf{s}\in\hat{S}_{b,l,j}} \mathbf{z}^{\mathbf{s}}_{l-1}\phi^{\mathbf{s}}_{l,j} \,. \tag{3}$$

We first consider a setup where all layers prior to l are frozen, i.e. earlier weights and consequently earlier activation values are not allowed to change. Note that this is the true situation at the first hidden layer only. Let all symbols (including sample sets) reflect the values *prior to* the weight update. Then it can be shown that any single sample $\mathbf{t}$ will be included in the sample cluster if

$$\mathbf{z}^{\mathbf{t}}_{l-1}\cdot\mathbf{w}_{l,j} > \eta \sum_{\mathbf{s}\in\hat{S}_{b,l,j}} \left(\mathbf{z}^{\mathbf{t}}_{l-1}\cdot\mathbf{z}^{\mathbf{s}}_{l-1}\right)\phi^{\mathbf{s}}_{l,j} \tag{4}$$

and removed otherwise. If we now define $\Sigma^{\mathbf{t}}_{l,j}$ as the net cost of the sample set at node j (layer l) as aligned with vector $\mathbf{z}^{\mathbf{t}}_{l-1}$, that is

$$\Sigma^{\mathbf{t}}_{l,j} = \sum_{\mathbf{s}\in\hat{S}_{b,l,j}} \left(\mathbf{z}^{\mathbf{t}}_{l-1}\cdot\mathbf{z}^{\mathbf{s}}_{l-1}\right)\phi^{\mathbf{s}}_{l,j} \,, \tag{5}$$

we can derive the precise conditions for which a sample $\mathbf{t}$ not previously in the cluster will be added:

$$\mathbf{z}^{\mathbf{t}}_{l-1}\cdot\mathbf{w}_{l,j} \leq 0; \qquad \Sigma^{\mathbf{t}}_{l,j} < 0; \qquad \left|\mathbf{z}^{\mathbf{t}}_{l-1}\cdot\mathbf{w}_{l,j}\right| < \eta\left|\Sigma^{\mathbf{t}}_{l,j}\right| \,, \tag{6}$$

or a member sample $\mathbf{t}$ (previously in the cluster) removed:

$$\mathbf{z}^{\mathbf{t}}_{l-1}\cdot\mathbf{w}_{l,j} > 0; \qquad \Sigma^{\mathbf{t}}_{l,j} > 0; \qquad \left|\mathbf{z}^{\mathbf{t}}_{l-1}\cdot\mathbf{w}_{l,j}\right| \leq \eta\left|\Sigma^{\mathbf{t}}_{l,j}\right| \,. \tag{7}$$

The absolute value signs are used here to emphasise that it is the magnitude of the values that is important. In effect, a margin is created around the decision boundary (where $\mathbf{z}^{\mathbf{t}}_{l-1}\cdot\mathbf{w}_{l,j} = 0$), with the size of this boundary directly specified by the learning rate and the summed loss of all samples that activate the specific node. Only samples falling within this boundary will either be drawn in or excluded from the sample set during the update. (This concept is illustrated in Figure 8 in the Appendix.) Note that this boundary *estimates* the net win of including or excluding additional samples by measuring their alignment with all other loss-generating samples in the set. If the effect is not as anticipated, the boundary will be shifted again in the following update. In this process, the boundary forms by separating samples that do not have coherent loss behaviour. Also note that a larger loss value implies that Equation 4 will be less strict, and, thus, samples that are less aligned with the weight vector might be accepted in cases where this would not have happened with a lower node-specific loss.

This process creates natural boundaries in the input space, separating areas that will benefit from being modelled separately. We illustrate this with an example: synthetic 1-dimensional data is generated, matching an underlying distribution as illustrated in Figure 1. A small network (1 hidden layer of 30 nodes)

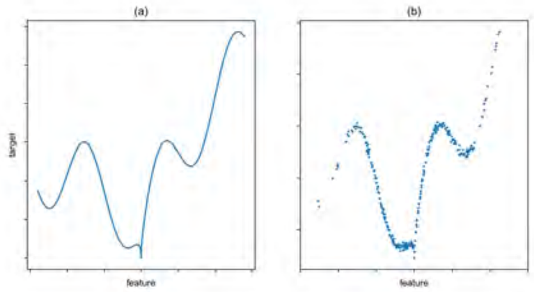


Fig. 1. 1-dim. regression example: (a) underlying distribution, (b) generated data.

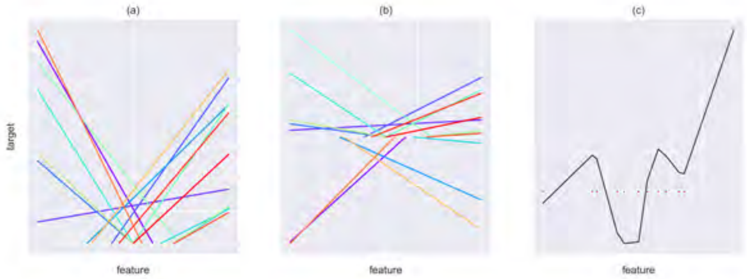


Fig. 2. 1-dim. regression example: (a) fan-in weight vectors, (b) vectors after activation and scaling, (c) weighted sum at target. Red dots indicate sample-set boundaries.

is trained to solve the regression task, and the resulting weight vectors are plotted as shown in Figure 2. Each fan-in vector at a node can be depicted as a line, based on the weight and bias value at that node. After ReLU-activation, all negative values are suppressed, and in the next layer, values are scaled based on fan-out weights. Finally, all contributions are summed in order to estimate the target value.

Once the sample-set boundaries have been drawn, scaling intermediary results to solve the overall task becomes a straightforward process; finding these boundaries is not. We propose that the heart of the training process can be studied through this clustering process of grouping relevant samples in the context of relevant features.

4 Experimental approach

Having provided our motivation for studying sample sets, we now outline our approach to investigating the dynamics of sample sets during training. We are interested in the size of the sample sets, the fraction of samples of a given class that is included in a sample set, how predictable sample-set membership is given class identity, and how informative sample-set membership is of class identity. These are intuitive concepts that we formalise below.

4.1 Setup

We train several networks, with the number of layers (excluding the input layer) being either 2, 5, or 9, and the number of nodes per hidden layer being either 20, 80, or 320. Bias parameters are added to the first hidden layer of each network. We use ReLU activation functions for the hidden layers. All networks are trained for 200 epochs on MNIST using the Adam optimisation algorithm [16] with a minibatch size of 60. Since we do not aim to maximise the performance of any of the networks, we train all the networks with the default optimiser hyperparameters suggested in [16] ($\alpha = 0.001, \beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 10^{-8}$). We train the networks with identity activation functions at the output layer and mean-squared-error loss. We repeat the training with cross-entropy loss (softmax activation functions followed by negative log-likelihood loss). The training is performed for 3 different initialisation seeds. For all networks, the training converges.

4.2 Measurements

We measure different characteristics of sample sets during training. The per-class *sample set* of node i in hidden layer l for class c, and the entire sample set of the same node are, respectively,

$$\hat{S}^{(c)}_{l,i} = \left\{\mathbf{s} : z^{\mathbf{s}}_{l,i} > 0, \mathbf{s} \in S_c\right\} \; ; \qquad\qquad \hat{S}_{l,i} = \bigcup_{c \in C} \hat{S}^{(c)}_{l,i} \; , \tag{8}$$

where S_c is the set of samples belonging to class c, and $z^{\mathbf{s}}_{l,i}$ is the activation of node i in hidden layer l for sample $\mathbf{s}$ [3,4,26]. At several points in the training process, we calculate the per-class *sample-set size* $|\hat{S}^{(c)}_{l,i}|$ for every node i in every hidden layer l for each class c.

We refer to the fraction of samples of a given class that is included in a sample set as the class-specific *activation fraction*, and define it for node i in hidden layer l for class c as

$$f^{(c)}_{l,i} = \frac{\left|\hat{S}^{(c)}_{l,i}\right|}{|S_c|} \; . \tag{9}$$

We can view the activation of node i in hidden layer l as a random variable $Z_{l,i}$, defining $\hat{Z}_{l,i} = 0$ if $Z_{l,i} \leq 0$ and $\hat{Z}_{l,i} = 1$ if $Z_{l,i} > 0$. If we also view the class of the input sample as a random variable Y, then we can approximate

$$P\left(\hat{Z}_{l,i} = 0, Y = c\right) \approx e_{l,i}(0,c) = \frac{|S_c| - \left|\hat{S}^{(c)}_{l,i}\right|}{|S|} \; ; \tag{10}$$

$$P\left(\hat{Z}_{l,i} = 1, Y = c\right) \approx e_{l,i}(1,c) = \frac{\left|\hat{S}^{(c)}_{l,i}\right|}{|S|} \; , \tag{11}$$

where S is the set of all samples. We approximate the conditional entropy of $\hat{Z}_{l,i}$ given Y and that of Y given $\hat{Z}_{l,i}$, for nodes with $|\hat{S}_{l,i}| \neq 0$ and $|\hat{S}_{l,i}| \neq |S|$, as

$$H\left(\hat{Z}_{l,i}\,|\,Y\right) \approx -\sum_{c\in C}\sum_{\hat{z}\in\{0,1\}} e_{l,i}(\hat{z},c)\log\left(\frac{e_{l,i}(\hat{z},c)}{\sum\limits_{x\in\{0,1\}} e_{l,i}(x,c)}\right); \tag{12}$$

$$H\left(Y\,|\,\hat{Z}_{l,i}\right) \approx -\sum_{\hat{z}\in\{0,1\}}\sum_{c\in C} e_{l,i}(\hat{z},c)\log\left(\frac{e_{l,i}(\hat{z},c)}{\sum\limits_{x\in C} e_{l,i}(\hat{z},x)}\right), \tag{13}$$

where we set $0\log(0) = 0$ [20]. Based on this, we define the *predictability* of node i in hidden layer l as

$$p_{l,i} = 1 + \frac{1}{\log(2)}\sum_{c\in C}\sum_{\hat{z}\in\{0,1\}} e_{l,i}(\hat{z},c)\log\left(\frac{e_{l,i}(\hat{z},c)}{\sum\limits_{x\in\{0,1\}} e_{l,i}(x,c)}\right), \tag{14}$$

and the *informativeness* as

$$u_{l,i} = 1 + \frac{1}{\log(|C|)}\sum_{\hat{z}\in\{0,1\}}\sum_{c\in C} e_{l,i}(\hat{z},c)\log\left(\frac{e_{l,i}(\hat{z},c)}{\sum\limits_{x\in C} e_{l,i}(\hat{z},x)}\right), \tag{15}$$

where we again set $0\log(0) = 0$ [20]. We do not define $p_{l,i}$ or $u_{l,i}$ for nodes with $|\hat{S}_{l,i}| = 0$ or $|\hat{S}_{l,i}| = |S|$, that is, for nodes that are always inactive (dead nodes) or always active (bias nodes). The predictability of a node is the difference between the maximum possible entropy of $\hat{Z}_{l,i}$ (based on the number of possible values of $\hat{Z}_{l,i}$) and the entropy of $\hat{Z}_{l,i}$ given Y, expressed as a fraction of the maximum possible entropy of $\hat{Z}_{l,i}$. Informally, this is proportional to the average amount of information known about $\hat{Z}_{l,i}$ given only an observation of Y. The informativeness indicates similar information about Y given $\hat{Z}_{l,i}$.

We calculate $f_{l,i}^{(c)}$, $p_{l,i}$, and $u_{l,i}$ at several logarithmically spaced points in the training process on a validation set of 12000 samples. We then aggregate these values as follows:

$$f_{l,i} = \frac{1}{|C|}\sum_{c\in C} f_{l,i}^{(c)}\,; \qquad f_l = \frac{1}{\left|N_a^{(l)}\right|}\sum_{i\in N_a^{(l)}} f_{l,i}\,; \tag{16}$$

$$p_l = \frac{1}{\left|N_b^{(l)}\right|}\sum_{i\in N_b^{(l)}} p_{l,i}\,; \qquad u_l = \frac{1}{\left|N_b^{(l)}\right|}\sum_{i\in N_b^{(l)}} u_{l,i}\,, \tag{17}$$

where $N_a^{(l)}$ is the set of nodes in hidden layer l for which $|\hat{S}_{l,i}| \neq 0$, and $N_b^{(l)}$ is the set of nodes in hidden layer l for which $|\hat{S}_{l,i}| \neq 0$ and $|\hat{S}_{l,i}| \neq |S|$.

5 Results

Here, we highlight interesting patterns observed by presenting results that display typical behaviour. Any other results that do not follow these patterns are pointed out. The performance of all networks during training is similar to that shown in Figure 3. All networks with 80 or more nodes per hidden layer achieve error rates smaller than 5% on the validation set. Those with 20 nodes per hidden layer achieve error rates smaller than 7% on the validation set.

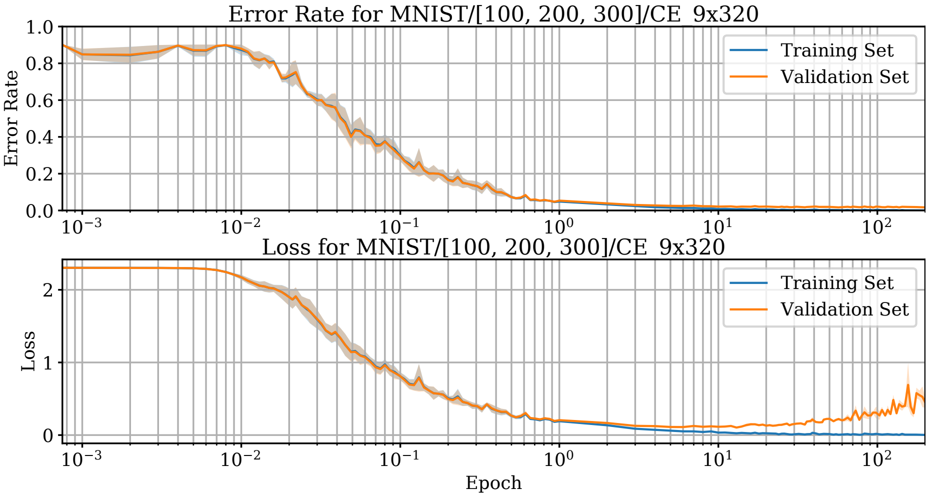


Fig. 3. Performance of 9×320 networks trained with cross-entropy loss, averaged across initialisation seeds. Shaded areas indicate the standard error.

5.1 Activation fraction

The activation fraction $f_{l,i}$ of the hidden nodes of a 9-layer network with 320 nodes per layer is shown in Figure 4. The average activation fraction f_l for the same architecture is shown in Figure 5. It is worth noting that the number of samples per class in the validation set is approximately constant across classes. As a result, $f_{l,i}$ is approximately equal to the fraction of *all* samples for which node i in layer l activates.

For most nodes, an increase in the sample-set size is observed very early in training. This indicates that the training process initially exposes most nodes to most samples. The nodes for which this does not hold activate for almost no samples during the entire training process, and, therefore, these nodes contribute very little to the network. The initial increase in sample-set size is followed by a gradual decrease for the rest of training. For networks trained with cross-entropy loss, this decrease is more rapid in shallower layers than in deeper layers. For

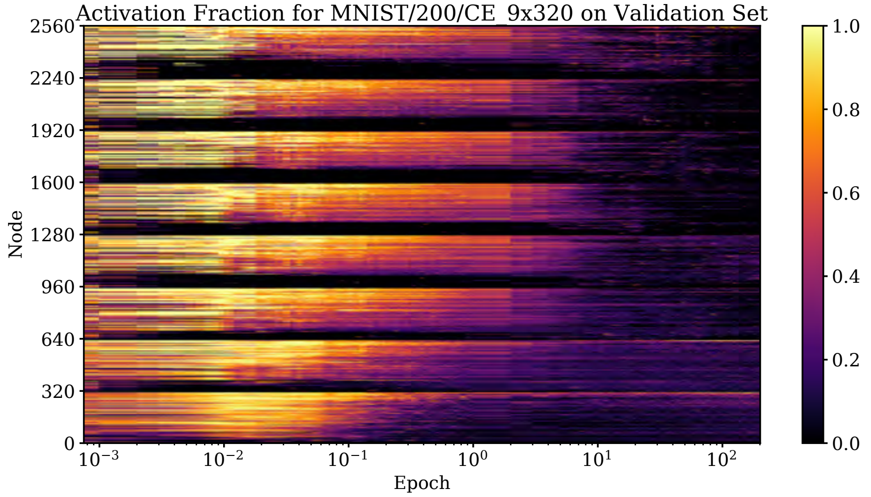


Fig. 4. Activation fraction for all hidden nodes in a 9×320 network trained with cross-entropy loss, calculated on the validation set. Nodes are numbered from shallowest to deepest.

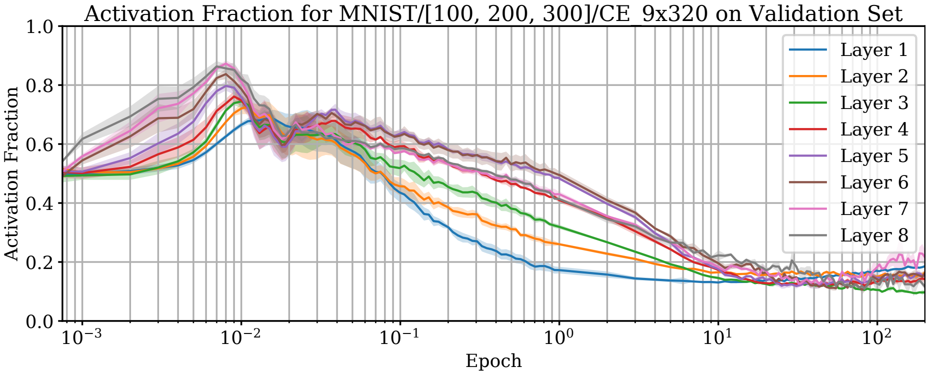


Fig. 5. Activation fraction averaged across nodes and initialisation seeds of 9×320 networks trained with cross-entropy loss, calculated on the validation set. Shaded areas indicate the standard error of the average across seeds. Layers are numbered from shallowest to deepest.

networks trained with MSE loss, this decrease is more rapid in the first hidden layer than in deeper layers.

Some results of individual networks are worth pointing out. For 9-layer networks trained with MSE loss, the sample sets of nodes in deeper layers contain almost all of the samples at the point where the sample sets are largest (see Figure 9 in the Appendix). For the 2-layer networks with 320 nodes per layer trained with MSE loss, the average activation fraction decreases during the entire training process (see Figure 10 in the Appendix). All other networks follow the same trends as in Figures 4 and 5.

5.2 Predictability

The average predictability p_l for a 9-layer network with 320 nodes per layer is shown in Figure 6. The average predictability of all layers increases at the start of training. For a few of the shallower layers, the predictability remains constant or decreases towards the end of training; for the rest of the layers, it continues to increase. The predictability of deeper layers is consistently higher than that of shallower layers. Not all results show the peaks that appear in Figure 6, and, therefore, we refrain from assigning meaning to them. These trends indicate that nodes, especially those in deeper layers, become increasingly more sensitive to class membership as training progresses. It also shows that nodes in deeper layers are more sensitive to class membership — a result that is confirmed by [4].

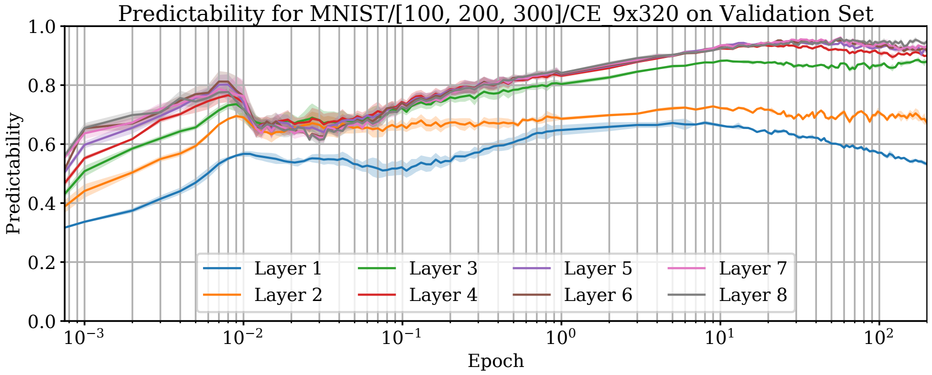


Fig. 6. Predictability averaged across nodes and initialisation seeds of 9×320 networks trained with cross-entropy loss, calculated on the validation set. Shaded areas indicate the standard error of the average across seeds. Layers are numbered from shallowest to deepest.

5.3 Informativeness

The average informativeness u_l for a 9-layer network with 320 nodes per layers is shown in Figure 7. For all layers, the informativeness increases and subsequently

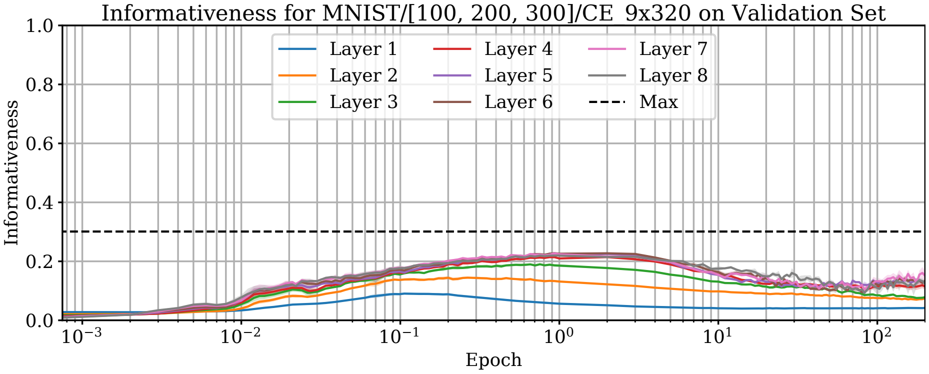


Fig. 7. Informativeness averaged across nodes and initialisation seeds of 9×320 networks trained with cross-entropy loss, calculated on the validation set. Shaded areas indicate the standard error of the average across seeds. Layers are numbered from shallowest to deepest. 'Max' indicates the maximum informativeness that can be achieved.

decreases slightly during training. Assuming that the 10 MNIST classes are perfectly balanced in the validation set, the maximum informativeness that can be achieved is $^{\log(2)}/_{\log(10)} \approx 0.3$, since the binary activation $\hat{Z}_{l,i}$ of a node can only have one of 2 values, but the class Y can have one of 10 values. The increase in informativeness shows that binary node activations become more indicative of class membership as training progresses. However, the amount of information required to establish the class with certainty means that the binary activation of any single node remains a poor predictor of class membership.

6 Discussion

To summarise our empirical findings: (1) Sample clusters initially increase in size and then decrease as training progresses. (2) Most nodes are exposed to most samples very early in the training process. (3) The point where the activation fraction starts decreasing overlaps with the point where network loss starts decreasing. (4) The size of sample clusters in the first layer decreases more rapidly than in deeper layers. (5) Binary node activations, especially of nodes in deeper layers, become more sensitive to class membership as training progresses. (6) Nodes in shallower layers tend to be less sensitive to class membership than nodes in deeper layers. (7) Binary node activations become slightly more indicative of class membership as training progresses, but remain poor predictors of class membership, even if accurate when applied as a group.

How do the above findings shed light on the creation of the sample clusters described in Section 3? When considering the training process, specific phases become evident: Upon initialisation, weights tend to point in arbitrary directions, and nodes in the network activate for samples found somewhere in the vicinity of their fan-in weight vector, creating the initial clusters. Since the loss is initially very large, additional samples are quickly drawn into the sample set of each of the nodes, according to Equation 4. It is during this time that the activation fraction grows. (See Figure 5.) This process continues up to the point where the activation fraction reaches its maximum. At this point, most nodes are being exposed to most samples, and also to most features.

It is only as the loss starts decreasing (Figure 3) that the activation fraction also starts decreasing, with nodes becoming increasingly specific. As nodes become more specific, weights become attuned to solving smaller subtasks, specifically trying to address any unresolved samples in its own sample set. At this point, nodes actively start selecting features (and directions in feature space) that are the most appropriate for solving its own subtask.

This process continues up to convergence, occasionally displaying a ripple effect where we conjecture that clusters are being re-shuffled. During training, only samples that have not been fully resolved contribute to weight updates. Samples with zero loss are simply ignored, unless a change in the network increases the loss value for such a sample as a side effect, which may cause clusters to break apart or re-combine.

Taken together, this process describes a practical approach to solving a high-dimensional clustering task, and specifically, to finding combinations of samples and features that show coherent behaviour and can therefore be modelled together effectively. It is only in the first layer that the raw input features are used to form clusters; in later layers, this clustering occurs in the transformed space produced by the previous layer.

While the findings presented at the start of this section were empirically confirmed across different architectures, the above interpretation is currently to be considered conjecture, rather than fact. In our current work we are analysing each of these statements in more detail.

7 Conclusion

Motivated by the role that sample sets play in the SGD training process, we studied the evolution of sample sets throughout training for several ReLU-activated networks. Our experiments reveal a consistent training process, as summarised at the start of Section 6. We provide some insight into the SGD training process by interpreting these findings using the conditions under which samples are included or excluded from sample sets (Section 3), and by discussing how this could relate to a high-dimensional clustering process (Section 6).

The current analysis is restricted to ReLU-activated networks. As the analysis in [4], which also studied the binary behaviour of individual nodes, was successfully extended to sigmoid-activated networks, we expect to be able to extend this study to a more diverse set of architectures and datasets as well. Although we do not directly address the apparent generalisation ‘paradox’ or improve the training process, the presented analyses and interpretations shed light on the training process from an interesting perspective.

References

1. Arpit, D., Jastrzebski, S., Ballas, N., Krueger, D., Bengio, E., Kanwal, M.S., Maharaj, T., Fischer, A., Courville, A.C., Bengio, Y., Lacoste-Julien, S.: A closer look at memorization in deep networks. In: Proceedings of the 34th International Conference on Machine Learning. pp. 233–242 (2017)
2. Bartlett, P.L., Mendelson, S.: Rademacher and gaussian complexities: Risk bounds and structural results. Journal of Machine Learning Research **3**, 463–482 (2002)
3. Davel, M.H.: Activation gap generators in neural networks. In: Proc. of the South African Forum for Artificial Intelligence Research FAIR. pp. 64–76. Cape Town, South Africa (12 2019)
4. Davel, M.H., Theunissen, M.W., Pretorius, A.M., Barnard, E.: DNNs as layers of cooperating classifiers. In: Thirty-Fourth AAAI Conference on Artificial Intelligence (2020)
5. Dinh, L., Pascanu, R., Bengio, S., Bengio, Y.: Sharp minima can generalize for deep nets. In: Proceedings of the 34th International Conference on Machine Learning. pp. 1019–1028 (2017)
6. Elsayed, G.F., Krishnan, D., Mobahi, H., Regan, K., Bengio, S.: Large margin deep networks for classification. In: Conference on Neural Information Processing Systems (2018)
7. Gale, T., Elsen, E., Hooker, S.: The state of sparsity in deep neural networks. arXiv Preprint **arXiv:1902.09574** (2019)
8. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning, pp. 22–25. MIT Press (2016)
9. Hardt, M., Recht, B., Singer, Y.: Train faster, generalize better: Stability of stochastic gradient descent. In: Proceedings of The 33rd International Conference on Machine Learning. pp. 1225–1234 (2016)
10. Hastie, T., Tibshirani, R., Friedman, J.: Chapter 2: Overview of supervised learning. In: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, pp. 9–41. Springer, 2 edn. (2017)
11. Hastie, T., Tibshirani, R., Friedman, J.: Chapter 7: Model assessment and selection. In: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, pp. 219–259. Springer, 2 edn. (2017)
12. Hochreiter, S., Schmidhuber, J.: Flat minima. Neural Computation **9**(1), 1–42 (1997)
13. Jiang, Y., Krishnan, D., Mobahi, H., Bengio, S.: Predicting the generalization gap in deep networks with margin distributions. In: International Conference on Learning Representations (2019)
14. Kawaguchi, K., Kaelbling, L.P., Bengio, Y.: Generalization in deep learning. In: Mathematics of Deep Learning. Cambridge University Press, to be published
15. Keskar, N.S., Mudigere, D., Nocedal, J., Smelyanskiy, M., Tang, P.T.P.: On large-batch training for deep learning: Generalization gap and sharp minima. In: International Conference on Learning Representations (2017)
16. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: International Conference on Learning Representations (2015)
17. Krueger, D., Ballas, N., Jastrzebski, S., Arpit, D., Kanwal, M.S., Maharaj, T., Bengio, E., Fischer, A., Courville, A.C.: Deep nets don't learn via memorization. In: International Conference on Learning Representations (2017)
18. Kuzborskij, I., Lampert, C.H.: Data-dependent stability of stochastic gradient descent. In: Proceedings of the 35th International Conference on Machine Learning. pp. 2815–2824 (2018)

19. Loroch, D.M., Pfreundt, F.J., Wehn, N., Keuper, J.: Sparsity in deep neural networks – An empirical investigation with TensorQuant. In: ECML PKDD 2018 Workshops. pp. 5–20 (2019)
20. MacKay, D.J.: Chapter 2: Probability, entropy, and inference. In: Information Theory, Inference, and Learning Algorithms, pp. 22–46. Cambridge University Press (2005)
21. Maurer, A., Pontil, M.: Structured sparsity and generalization. Journal of Machine Learning Research **13**, 671–690 (2012)
22. Neyshabur, B., Bhojanapalli, S., McAllester, D., Srebro, N.: Exploring generalization in deep learning. In: Conference on Neural Information Processing Systems (2017)
23. Novak, R., Bahri, Y., Abolafia, D.A., Pennington, J., Sohl-Dickstein, J.: Sensitivity and generalization in neural networks: an empirical study. In: International Conference on Learning Representations (2018)
24. Pinkus, A.: Approximation theory of the mlp model in neural networks. Acta Numerica **8**, 143–195 (1999)
25. Sokolić, J., Giryes, R., Sapiro, G., Rodrigues, M.R.D.: Robust large margin deep neural networks. IEEE Transactions on Signal Processing **65**(16), 4265–4280 (2017)
26. Theunissen, M.W., Davel, M.H., Barnard, E.: Insights regarding overfitting on noise in deep learning. In: Proc. of the South African Forum for Artificial Intelligence Research FAIR. pp. 49–63. Cape Town, South Africa (12 2019)
27. Theunissen, M.W., Davel, M.H., Barnard, E.: Benign interpolation of noise in deep learning. South African Computer Journal (12 2020)
28. Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O.: Understanding deep learning requires re-thinking generalization. In: International Conference on Learning Representations (2017)

A Illustration of Equation 6 and 7

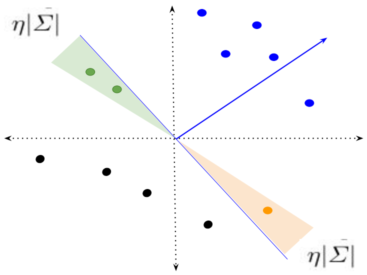


Fig. 8. The size of $\eta|\Sigma|$ determines the area where additional samples will be included in (green) or excluded from (orange) the sample set, after the next weight update. Weight vector is shown in blue, with current boundary line indicated, also in blue. The sample-set membership of other samples already in the sample set (blue) and outside the sample set (black) are not affected.

B Additional results

Figure 9 shows the activation fraction for networks in which the nodes in deeper layers have sample sets that contain almost all of the samples at the point where the sample sets are largest. This pattern holds for all 9-layer networks trained with MSE loss.

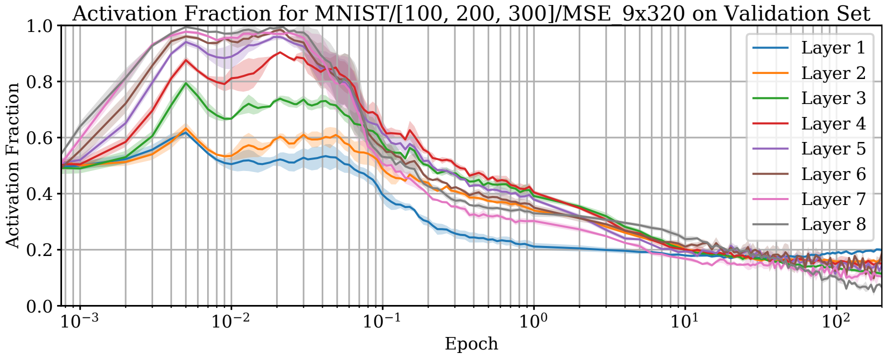


Fig. 9. Activation fraction averaged across nodes and initialisation seeds of 9×320 networks trained with mean-squared-error loss, calculated on the validation set. Shaded areas indicate the standard error of the average across seeds. Layers are numbered from shallowest to deepest.

Figure 10 shows the activation fraction for the networks in which an increase in average activation fraction is not observed. This only holds for 2-layer networks with 320 nodes per layer trained with MSE loss.

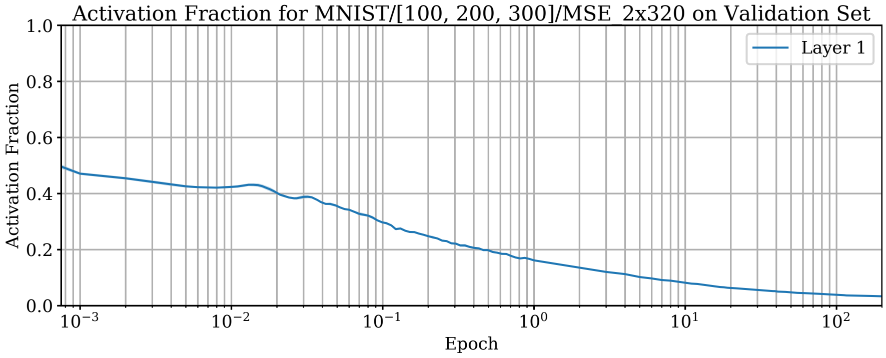


Fig. 10. Activation fraction averaged across nodes and initialisation seeds of 2×320 networks trained with mean-squared-error loss, calculated on the validation set. Shaded areas indicate the standard error of the average across seeds.

Using a meta-model to compensate for training-evaluation mismatches

Dylan Lamprecht[1][0000−0002−2172−3991] and Etienne Barnard[2][0000−0003−2202−2369]

Multilingual Speech Technologies (MuST), North-West University, South Africa; and CAIR, South Africa.

Abstract. One of the fundamental assumptions of machine learning is that learnt models are applied to data that is identically distributed to the training data. This assumption is often not realistic: for example, data collected from a single source at different times may not be distributed identically, due to sampling bias or changes in the environment. We propose a new architecture called a meta-model which predicts performance for unseen models. This approach is applicable when several 'proxy' datasets are available to train a model to be deployed on a 'target' test set; the architecture is used to identify which regression algorithms should be used as well as which datasets are most useful to train for a given target dataset. Finally, we demonstrate the strengths and weaknesses of the proposed meta-model by making use of artificially generated datasets using a variation of the Friedman method 3 used to generate artificial regression datasets, and discuss real-world applications of our approach.

Keywords: Generalization · Meta-model· Mismatched distributions· Robustness· Machine learning· Tree-based models.

1 Background

The past decade has witnessed great successes in machine learning, in various areas [17], including image classification [3], speech recognition [13] and recently natural language processing [4].

This model can then be deployed, and its performance can be measured on a *test* or *evaluation set* - the essential characteristic of Machine Learning is that the training and evaluation sets are distinct, and the goal is to achieve good *generalization* on the evaluation set.

In classical learning theory, one assumes that the training and evaluation data are drawn from the same probability distribution [15]. However, that assumption is often not realistic for practical applications. For the sake of convenience, training data for an image recognition system may be collected in a particular location, whereas the system may be deployed (and evaluated) more widely. A more troubling example was recently reported for a system that predicts the probability that people are likely to re-offend in the USA criminal justice system: the algorithms show an implicit bias when predicting for two different ethnic groups, because it had been trained with biased data [16]. The current assumption that is required for the analysis of machine learning models is that models are built such that the dataset the model trains on and the set which the model is tested against have the same statistical distribution [11]. This research wishes to identify statistically similar, but not identical distributions, and show that good machine learning performance can be achieved under those conditions.

Our goal is to demonstrate the meta-model's ability to identify the presence of a mismatch between datasets as well as identify which datasets have the least change in distribution. This scenario is important in several applications of pattern recognition, such as speech recognition where clean training data must be used to recognize noisy speech, or computer vision when simulated images are used as training for real-world recognition.

We are interested in problems where this mismatch is unavoidably significant, namely in applications where several relevant 'proxy' training sets are available, and none of these is guaranteed to match the 'target' evaluation set. Such tasks are common in certain sectors of the financial industry, where several prepackaged databases may be available and the goal is to use one or more of those for prediction on a new dataset. Since our simulated data are based on a regression task from that industry, we only show results for a particular family of regression models. However, it will become clear that the same approach can be applied to more general tasks.

2 Related work

Generalization is the ability for machine learning algorithms to perform well on a set of unseen samples. This ability can be enhanced by giving additional information to sets of training data. The method frequently used to improve invariance to change is to make use of the transformation of a dataset [7]. The rationale behind a linear transformation is that as long as a model can adapt to a linear transformation without a drop in performance, the model can adapt to realistic changes and hence can generalize better. Specifically, one focuses on determining how to compensate for a covariate shift. However, this does not allow for a shift in the target variable or the relationships between the features and the target variables.

In our context, it is not clear how one can ensure that the generated data approximates the distribution of the evaluation dataset; thus, this approach will generally bias the model to predict on the generated data.

To compensate for this, Generalized robust risk minimization (GRRM) creates an ensemble of training sets in order to have an increase in generalization performance. GMMR is a variation of Robust risk minimization (RRM), which is a robust variation of Empirical risk minimization (ERM) [11]. This model makes use of aggregating ensembles of training data. Since ERM drastically degrades when noise is introduced [11], GRRM is a method to improve the generalization performance of models as demonstrated in its ability to adapt to noise, covariate shifts and weak multi-labels [8].

The most frequent method for measuring differences in distributions prior to running them is by making use of Distance Metric Learning. Distance Metric Learning (DML) is an important method to use as a baseline approach in many machine learning problems. This quantifies the difference in distributions between the two datasets ('training' and 'evaluation' or 'proxy' and 'target'). This makes the algorithms adaptable to changes and identifies which datasets are distributed more closely to one another [2]. Various methods of utilizing distance metrics have been proposed – as an example, we discuss Consistent distance metric learning (CDML) [2].

To compensate for variation between datasets, CDML [2] makes use of distance metrics to balance the cost function of a dataset by making use of feature importance pairs, using importance sampling methods from supervised learning for metric learning. Two important weighting strategies were tested:

- Estimating the importance weights of data instances before calculating importance weights for instance pairs.
- Estimating the importance of instance pairs directly.

This method has been shown to be effective for classification tasks on clustering problems for real-world and artificial datasets [2].

We have reviewed several methods that have been proposed to deal with mismatches between training and test data. Although methods of making models more robust are useful when dealing with mismatches between datasets, they rely on each dataset being similar to the target set, while our proposed method focuses on measuring which datasets are the closest related to a target dataset.

3 Methodology

We now describe our approach to the mismatch problem, which employs a metalmodel. The purpose of the meta-model is to use selected features from trained models to predict the performance of the model on an unseen dataset. This is particularly useful since it provides insight into which datasets and regression algorithms will perform well, without a validation set, as is required in the case when the datasets are distributed significantly differently.

3.1 Algorithm overview

The algorithm trains a set of models using various combinations of proxy data, target data and regression algorithms as outlined in Figure 1. For each trained model, we measure various 'meta-features' and then create a meta-model which predicts the accuracy of a particular proxy-target-algorithm combination based on those meta-features. This meta-model can then be used to predict which proxy and regression algorithm should be used for a given target set.

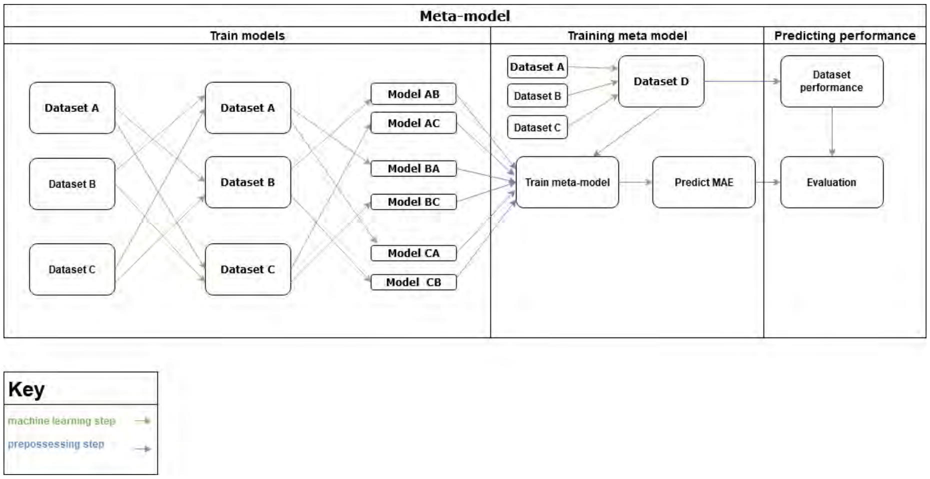


Fig. 1. Flow diagram of process for meta-model training and evaluation. During the training models phase, we collect the meta-data used to train the meta-model by forming each possible test-train pair of datasets and train models for each test-train combination. We then proceed to the process summarized in 'Training meta model', where we process all the features (outlined in Subsection 3.3) of the models trained in the 'train models' phase as meta-data. We use the meta-data as features to train the meta-model to predict the mean absolute error (MAE) for a given model. As a test set, we train each model using each other dataset and applied it to the features of an unseen test dataset to create the 'meta-features'. We then compare the actual MAE to the predicted MAE of the unseen test set to evaluate the performance of the meta-model.

3.2 Training and using the meta-model

Training the models To collect relevant training data for the meta-model (meta-data), we trained a variety of models and extracted the relevant information (as described in Subsection 3.3) on the results of the training and test sets. For each dataset, each other dataset was used as a training set to form a test-train pair. There were six regression algorithms used to train six models for each dataset combination specified in Subsection 3.4 'Input to the model'.

Training the meta-model To train the meta-model we made use of the meta-data of models and tested the results against the target dataset. The process of training the meta-model consisted of using the meta-data as features (as described in Subsection 3.3) and using the mean absolute error (MAE) as the target feature. We then trained a model using the catboost regression algorithm to predict the MAE of a model tested on an unseen dataset. To test the performance, we used a dataset not used in the training set and predicted performance of each dataset and each model on this unseen dataset.

3.3 Meta-model features

To create a meta-model, we measured a variety of metrics of the trained models, related to predicted ranges, distance metrics and general information about each of the features and then predicted the MAE for each model by using the following classes of features:

- 'Meta-features' about the features used for training the underlying models: these include the bins (outlined in the next paragraph) of each of the features in the training and test sets, their means and standard deviations, and several distance metrics between the pair of datasets (Kullback–Leibler divergence [12], Kolmogorov–Smirnov test [5], Hellinger distance [6] and Wasserstein distance [14])
- 'Meta-features' about the predicted target of the underlying models: bins (outlined in the next paragraph) of the predicted values for both sets, their means and standard deviations, and the sum of the predicted values.

Bins We created twelve bins, which is done by sorting values of a feature or predicted target in ascending, dividing them into equal parts and then add a value representative of the interval each value falls under. Then for each feature and predicted target we had an equal number of entries for all the features for both training and test set and used the boundary of each bin as features. Hence, these features are the relevant percentiles of the features for the two sets.

3.4 Regression algorithms

Various tree-based methods (Catboost, Extra tree regressor, Gradient boosted regressor, Light GBM, Random forest and XGBoost) were employed as regression algorithms. These tree-based regressors were chosen and parameters were chosen from literature [10] along with state-of-the-art variations of gradient boosted trees [1]). Based on informal experiments, we believe that these are reasonable parameter choices for each of the algorithms. Since our aim is to estimate the ranking of these algorithms, rather than deriving a single best algorithm, it is not necessary to obtain the very best setting for any of the parameter sets. It is also evident that the performance of the algorithms is relatively similar as demonstrated in table 4

4 Experiments

We investigate the performance of the proposed meta-model approach by showing the results obtained on a group of artificial datasets. We generated related datasets by using the mathematical function

$$tan^{-1}\left(\frac{x_2x_3 - \frac{1}{x_2x_4}}{x_1}\right) \tag{1}$$

as proposed by Friedman [1] (his method '3'). This method was chosen since it was the method that generated datasets most closely resembling the real-world data (which we cannot share due to confidentiality constraints) this method was originally developed for. For the related datasets we wish to demonstrate how the meta-model adapts to the following situations:

- Generating noise: We generated a uniform distribution between 0 and 1 ($U(0,1)$), multiplied the result with the noise factor and then added it to step three in Subsection 4.1 of the Friedman 3 implementation. For each dataset we added the amount of noise equal to the numeric order of the dataset to each dataset. Here we test how robust the meta-model is to minor changes in data.

$$tan^{-1}\left(\frac{x_2x_3 - \frac{1}{x_2x_4}}{x_1}\right) \times N(0,1) \times c \times dataset\ number \tag{2}$$

- Scale changes: We scaled each of the features generated in step 1 to create several unique datasets. For example, if the scale change was five and there are five datasets, the datasets would have the following ranges for the features [(0,0.2), (0.2,0.4), (0.4,0.6), (0.6,0.8), (0.8,1)]. Here we are primarily testing for whether the meta-model can identify which datasets are relatively close to each target dataset in terms of scale changes.

[1] The code is available on Github, under the project name meta-model.

- Data shifts: We multiplied the dataset number with data shift factor and added the result to step three in the Friedman 3 implementation.

$$tan^{-1}\left(\frac{x_2x_3 - \dfrac{1}{x_2x_4}}{x_1}\right) + c \times dataset\ number \tag{3}$$

- A combination of scale changes and data shifts: We combined the dataset from both scale changes and data shifts, in order to study how robust the meta-model was to combined changes.
- Model selections: We ran the model with different underlying regression algorithms. This was done as a method to determine which algorithms performed better across datasets with a scale change of 2.
- Dataset size: We tested how well the meta-model performs when the underlying datasets are trained on different numbers of underlying elements, to test how many samples are required in a dataset for the meta-model to be able to estimate performance well.

This will probe the meta-model's ability to generalize, by demonstrating the effects that different data alterations have on the meta-model and how the meta-model adapts to change. This will be used as a proxy for how real-world datasets are differently distributed and how the meta-model adapts to the changes.

4.1 Friedman 3 implementation

We generated 10,000 elements for each dataset, each element consisting of four features x_1, x_2, x_3 and x_4 as well as a target. To generate the features we implemented the following steps:

1. For each feature, we uniformly generated a value between 0 and 1. For each dataset, we generated the features with the seed used to initialize the random-number generator set equal to the number of the dataset (e.g. dataset 0 generates on seed 0, dataset 1 generates on seed 1 etc.)
2. Then we applied the following transformations to each feature
   - $x_1 = feature\ 1 \times 100$
   - $x_2 = feature\ 2 \times 520\pi + 40\pi$
   - $x_3 = feature\ 3$
   - $x_4 = feature\ 4 \times 10 + 1$
3. We finally applied the following transformation to get the target for each element:

$$target = tan^{-1}\left(\frac{x_2x_3 - \dfrac{1}{x_2x_4}}{x_1}\right) \tag{4}$$

4.2 Metrics measured

To evaluate the performance of the trained meta-model, we made use of the following metrics.

- Regression performance: We compared the true MAE to the predicted MAE using the correlation coefficient. The reason we are using correlation coefficient instead of a standard MAE or R^2 is since the modifications change the target values, which would indicate a larger error in terms of value in many cases, but because the emphasis of this paper is to find similarity, the relative order is more important and the correlation coefficient is able to measure the relative value without being influenced as much by the changes in for example generative noise.
- Ranking datasets performance: By using the MAE, we set up two lists with the datasets ranked from the best performance to the worst performance, using the actual and the predicted MAE. Then we used the Kendall Tau test [9] to determine relative similarity (on a scale from -1 to 1) of the two rankings.

4.3 Results

We now report on the results that were obtained when datasets generated according to these protocols were processed with the algorithm described in Section 3. For each setting of the generation parameters, we measure the correlation coefficient between the actual and predicted MAEs for each dataset-regressor combination using Leave-one-out cross validation on each of the 11 datasets. We also evaluate the accuracy of the meta-model in predicting the relative rankings of the different dataset combinations, using the Kendall Tau test [9].

	10	20	50	100
Metric	Correlation coefficient	Correlation coefficient	Correlation coefficient	Correlation coefficient
Average	0.9827	0.9712	0.9529	0.9454
Standard deviation	0.0321	0.0466	0.0881	0.1002
Median	0.9981	0.9988	0.9983	0.9983
Metric	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau
Average	0.9515	0.9232	0.8828	0.8949
Standard deviation	0.0285	0.0797	0.1164	0.1094
Median	0.9667	0.9556	0.9111	0.9111

Table 1. Scale changes

	10	100	1000	10000
Metric	Correlation coefficient	Correlation coefficient	Correlation coefficient	Correlation coefficient
Average	0.8363	0.9261	0.9817	0.9857
Standard deviation	0.2848	0.1678	0.0217	0.0130
Median	0.9879	0.9923	0.9934	0.9907
Metric	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau
Average	0.6929	0.8101	0.9515	0.9475
Standard deviation	0.1233	0.1688	0.0464	0.0556
Median	0.7333	0.8222	0.9556	0.9556

Table 2. Dataset sizes

	0.01	0.02	0.05	0.1
Metric	Correlation coefficient	Correlation coefficient	Correlation coefficient	Correlation coefficient
Average	0.4478	0.5043	0.5987	0.6082
Standard deviation	0.6096	0.5610	0.4201	0.3957
Median	0.7372	0.7081	0.8010	0.7560
Metric	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau
Average	0.3778	0.3818	0.4869	0.4626
Standard deviation	0.5440	0.5123	0.3120	0.3428
Median	0.4667	0.5556	0.6000	0.6000

Table 3. Data shift

	Catboost	Light GBM	XGBoost	GBT	Random forest	Extra tree	all regression algorithms
Metric	Correlation coefficient	Correlation coefficient	Correlation coefficient	Correlation coefficient	Correlation coefficient	Correlation coefficient	Correlation coefficient
Average	0.9863	0.9886	0.9903	0.9594	0.9874	0.9860	0.9748
Standard deviation	0.0146	0.0119	0.0067	0.0449	0.0119	0.0132	0.0202
Median	0.9930	0.9934	0.9912	0.9833	0.9938	0.9880	0.9805
Metric	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau
Average	0.8909	0.8990	0.9071	0.8343	0.9394	0.9030	0.8454
Standard deviation	0.0608	0.0565	0.0464	0.1072	0.0850	0.0623	0.0542
Median	0.8667	0.8667	0.9111	0.8222	0.9556	0.9111	0.8441

Table 4. Regression algorithm selection

		10	20	50
Dataset section	Metric	Correlation coefficient	Correlation coefficient	Correlation coefficient
Scale changes	Average	0.9785	0.9752	0.9695
	Standard deviation	0.0191	0.0203	0.0395
	Median	0.9862	0.9829	0.9831
		Kendall Tau	Kendall Tau	Kendall Tau
Scale changes	Average	0.8459	0.8286	0.8277
	Standard deviation	0.0385	0.0664	0.0962
	Median	0.8381	0.8476	0.8476

Table 5. Combining data shifts and scale changes to measure the effects on scale changes

		10	20	50
Dataset section	Metric	Correlation coefficient	Correlation coefficient	Correlation coefficient
Data shifts	Average	0.7416	0.7054	0.7621
	Standard deviation	0.2924	0.4539	0.3241
	Median	0.8873	0.9094	0.8869
		Kendall Tau	Kendall Tau	Kendall Tau
Data shifts	Average	0.5749	0.5697	0.5905
	Standard deviation	0.0621	0.1166	0.0853
	Median	0.5810	0.6381	0.6000

Table 6. Combining data shifts and scale changes to measure the effects on data shifts

	0	0.01	0.02	0.05	0.1
Metric	Correlation coefficient	Correlation coefficient	Correlation coefficient	Correlation coefficient	Correlation coefficient
Average	0.9853	0.9855	0.9798	0.9616	0.9663
Standard deviation	0.0143	0.0152	0.0218	0.0454	0.0357
Median	0.9926	0.9921	0.9912	0.9819	0.9845
Metric	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau	Kendall Tau
Average	0.9273	0.9232	0.9192	0.8949	0.8505
Standard deviation	0.0637	0.0491	0.0623	0.0802	0.0999
Median	0.9111	0.9111	0.9111	0.9111	0.8667

Table 7. Additive noise

Processing	Elements	10 datasets	20 datasets	30 datasets	50 datasets
Single core processing	100	51.19	195.98	436.93	1,211.22
	1,000	92.88	367.83	788.92	2,152.32
	10,000	721.62	2,759.05	5,550.93	15,369.38
Multicore processing	100	8.98	28.65	63.04	174.33
	1,000	12.54	48.22	102.56	279.13
	10,000	86.44	316.97	690.90	1,898.71

Table 8. Computation

5 Discussion

The following findings were obtained from the experiments discussed in Section 4 outlined in tables 1- 8.

The meta-model is able to adapt to differences in scale changes, and larger-scale changes result in less accurate meta-model predictions. This is to be expected since the model is unable to reproduce the relationship completely; the greater the difference in the scale, the more significant the inability to reproduce the relationship is as shown in Table 1.

From Table 2 it is evident that the meta-model's performance is the inverse of the scale changes in Table 1 i.e. greater numbers of samples in the datasets result in better predictions of the MAE. The meta-model improves logarithmically with a greater number of elements in a dataset: for this task, we need at least 1,000 samples to achieve a correlation coefficient of better than 0.95.

By implementing data shifts and breaking the relationship between the features and targets in Table 3, the prediction performance of the meta-model is reduced significantly. As the data shift increases, the model is able to predict *relative* MAEs more accurately: although in absolute terms the MAE becomes worse as the data shift increases, the correlation coefficient increases since the meta-model responds more accurately to larger changes.

Table 4 shows the effect when the meta-model is trained on models with different underlying regression algorithms. We specifically looked at the regression algorithms [Catboost, Light GBM, XGBoost, gradient boosted tree, random forest, Extra tree regressor] with hyperparameters as mentioned in Subsection 3.4. In general, the majority of regression algorithms perform similarly, the only two noticeable exceptions are gradient boosted trees, which has a lower correlation coefficient as well as a lower Kendall Tau ranking, and the random forest algorithm is able to rank the models (using Kendall Tau) 0.03 better than models trained with any other regression algorithm.

Table 5 We combined scale changes with data shifts (a standard 0.01 data shift) to test how well the meta-model adapts to different types of changes in different datasets. For scale changes combining the results with the data shifts did hurt the Kendall Tau ranking by around 0.1 for each correlation. However, the correlation coefficient was less sensitive to scale changes in comparison to the results obtained for only using scale changes in Table 1. Since the dataset

now combines both scale changes and data shifts, the overall effect is somewhere intermediate to the effects of each of the separate modifications.

Table 6 For the data shifts when combining scale changes with data shifts (a standard 0.01 data shift), the meta-models did perform better in comparison to Table 3, where the correlation coefficient improved by around 0.3 and the Kendall Tau ranking improved by 0.2. This is probably due to the dataset containing the scale changes, which are datasets the meta-model could predict more successfully.

When comparing the effect of additive noise on a meta-model in Table 7, we again found the expected behaviour: since the noise breaks the correlation between features and targets, more accurate prediction is achieved at lower noise levels.

We demonstrate the computational requirements of the meta-model, by documenting the architecture's performance in Table 8, we used a Ryzen 2600X CPU (containing 6 CPU cores with 12 threads) and the time was measured in seconds. In general, the computational performance followed two trends, the first is that the process benefits greatly with the number of cores and that the computational performance scales with the number of cores. The second is that the computational performance between dataset sizes can be described using the formula below

$$Model_2 \approx Model_1 \left(\frac{Datasets_2}{Datasets_1}\right)^2 \tag{5}$$

Where:

$Model$ = The computation required to run the meta-model
$Datasets$ = The number of dataset used to train the meta-model

6 Conclusion

The meta-model has proven to be valuable in both providing an ordered ranking of the datasets and predicting how accurate each model would be for a given test dataset. For scale changes, we compared different levels of scale changes and the greater the level of scale change the worse the model performs. The meta-model can perform well with underlying models that are trained on very few elements; however, the performance of the meta-model improves with more elements in the underlying datasets. Creating a data shift does hurt model performance significantly; however, the level of change did not significantly affect the meta-model's performance. The meta-model's performance did not differ significantly when trained with different underlying regression algorithms, the only notable exception being gradient boosted trees. When applying scale changes with data shifts, the additional models harm the performance of the scale changes by a greater degree than the maximum scale change, and the data shifts perform better by a significant amount. Adding generated noise does harm the model's ability to predict, although it is able to compensate for the change for relatively small amounts of noise.

This method was originally adapted from a real world application on real world data (the nature of which can not be disclosed because of commercial confidentiality), and the datasets generated reflect the size and number of features in the real-world data. An interesting direction for future work would be to investigate combinations of datasets for training: we currently only predict the performance for each individual 'proxy' set, but optimal performance may well require combining several sets. Computing a meta-model prediction for each combination of sets is computationally prohibitive, so a more insightful approach is required.

In terms of deploying the system in practice, the architecture provides a summary of the predicted MAE's for each model and dataset for a given test set. From this, a user can select from a few models and use the predicted MAE as a proxy for the model's accuracy. This allows a user to set up a range of probable values for a given dataset along with a method to quantify the probability of each outcome.

References

1. Breiman, L.: Bagging predictors. Machine learning **24**(2), 123–140 (1996)
2. Cao, B., Ni, X., Sun, J.T., Wang, G., Yang, Q.: Distance metric learning under covariate shift. In: Twenty-Second International Joint Conference on Artificial Intelligence (2011)
3. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
4. Jordan, M.I., Mitchell, T.M.: Machine learning: Trends, perspectives, and prospects. Science **349**(6245), 255–260 (2015)
5. Lilliefors, H.W.: On the Kolmogorov-Smirnov test for normality with mean and variance unknown. Journal of the American statistical Association **62**(318), 399–402 (1967)
6. Lindsay, B.G.: Efficiency Versus Robustness: The Case for Minimum Hellinger Distance and Related Methods. Annals of Statistics **22**(2), 1081–1114 (06 1994)
7. Ling, J., Jones, R., Templeton, J.: Machine learning strategies for systems with invariance properties. Journal of Computational Physics **318** (05 2016). https://doi.org/10.1016/j.jcp.2016.05.003
8. Mazuelas, S., Perez, A.: General Supervision via Probabilistic Transformations. arXiv e-prints arXiv:1901.08552 (Jan 2019)
9. Noether, G.E.: Why Kendall Tau? Teaching Statistics **3**(2), 41–43 (1981)
10. Olson, R.S., La Cava, W., Mustahsan, Z., Varik, A., Moore, J.H.: Data-driven Advice for Applying Machine Learning to Bioinformatics Problems. arXiv e-prints arXiv:1708.05070 (Aug 2017)
11. Osama, M., Zachariah, D., Stoica, P.: Robust Risk Minimization for Statistical Learning. arXiv e-prints arXiv:1910.01544 (Oct 2019)
12. Perez-Cruz, F.: Kullback-Leibler divergence estimation of continuous distributions. In: 2008 IEEE International Symposium on Information Theory. pp. 1666–1670 (2008)
13. Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., Vesely,

K.: The Kaldi Speech Recognition Toolkit. In: IEEE 2011 Workshop on Automatic Speech Recognition and Understanding. IEEE Signal Processing Society (Dec 2011), iEEE Catalog No. CFP11SRW-USB

14. Vallender, S.S.: Calculation of the Wasserstein Distance Between Probability Distributions on the Line. Theory of Probability & Its Applications **18**(4), 784–786 (1974)
15. White, H.: Artificial neural networks: approximation and learning theory. Blackwell Publishers, Inc. (1992)
16. Yapo, A., Weiss, J.: Ethical Implications of Bias in Machine Learning (01 2018). https://doi.org/10.24251/HICSS.2018.668
17. Yu, A.W.: Effective and Efficient Learning at Scale. Ph.D. thesis, Carnegie Mellon University (2019)

Link Prediction in Knowledge Graphs using Latent Feature Modelling and Neural Tensor Factorisation

Luyolo Magangane and Willie Brink

Stellenbosch University, Stellenbosch, South Africa
luyolo.nqobile@gmail.com, wbrink@sun.ac.za

Abstract. Knowledge graphs can be used to represent interconnected facts about multiple domains as entities (nodes) and relations (edges). The resource description framework (RDF) formalism can be used to encode such facts as subject-predicate-object triples. Link prediction then powers knowledge discovery by scoring possible relationships between entities. Tensor decomposition is an attractive approach to link prediction, as relational domains are usually high-dimensional and sparse; a setting where factorisation methods, particularly the HypER model, have shown very good results. Modern approaches typically also contain nonlinear neural networks to enable the learning of powerful latent representations of entities and relations in a continuous vector space. We introduce optimisations to the training algorithm of HypER, by using batch normalisation to compensate for covariate shift caused by hypernetworks, and propose HypER+. We see similar performance to the HypER baseline on the WN18 dataset, and significant improvement on the FB15k dataset. We then extend our model by initialising entity and relation embeddings with pre-trained word vectors from the GloVe language model, and see further improvements over the baselines on the more challenging WN18RR and FB15k-237 datasets. Our results establish HypER+ as a state-of-the-art model in latent feature modelling based link prediction.

Keywords: Link prediction · Tensor decomposition · Hypernetworks.

1 Introduction

Reasoning over knowledge expressed in natural language is a problem at the forefront of artificial intelligence research. Question answering is a core task of this problem, and is concerned with giving machines the capability of generating an answer given a question.

Knowledge graphs (KGs) model facts as entities (nodes) and relations between entities (edges) [1]. The resource description framework formalism encodes facts as triples of the form "subject-predicate-object", where the subject and object are entities, and the predicate is a relation [2]. An example could be the fact that the subject "Chadwick Boseman" is related to the object "Black Panther" by the predicate “starred in”. Question answering then relies on knowledge discovery; a step-by-step deductive process of inferring new facts from a given set of known facts. Statistical relational learning (SRL) solves the knowledge discovery problem by constructing models with measures of uncertainty in plausible facts not contained in KGs [3].

There has been encouraging progress in SRL to model knowledge in open-domain settings [4]. Link prediction, i.e. inferring plausible relations between KG entities, is now often used as a paradigm for knowledge discovery [5–7]. Latent feature modelling using tensor factorisation [8] is an approach to link prediction that has seen some promising results [9–11]. The first major milestone was demonstrating the bilinear tensor product as a promising model to link prediction [12]. In this model a vector representation of the subject is multiplied with a predicate matrix to produce an entity-relational vector, and the inner product with an object vector then provides a relational plausibility score for the subject and object entities. The second milestone was an integration of the bilinear tensor product and neural networks [13], effectively extending linear tensor factorisation techniques to nonlinear techniques and also enabling the use of pre-trained word embeddings to initialise entity and relation vectors. Third was the use of complex valued embeddings for link prediction [14], to effectively capture antisymmetric relations. Up until that point research had focused on minimising the parameterisation of models. Convolutional networks are parameter efficient, and progress was again realised with the use of deep convolutional models for link prediction [15]. The HypER model [16] introduced the use of hypernetworks (a class of meta-networks trained to configure a main network [17]) to transform relational vectors into relation-specific convolutional filters used by a main network.

In this work we first introduce training algorithm optimisations to a TensorFlow reimplementation of a baseline neural tensor network (NTN) model [18]. We then adapt the HypER model by applying batch normalisation, in an effort to mitigate covariate shift in its hypernetwork, and call the new model HypER+. Finally, in an attempt to leverage semantic information from very large text corpora, we integrate pre-trained word vectors into the HypER+ training process to initialise entity and relational vectors in place of standard random initialisation. Our experimental results on benchmark datasets establish HypER+ as a state-of-the-art model in latent feature modelling based link prediction.

2 Related Work

Statistical relational learning is comprised of three paradigms: latent feature modelling, graph modelling and inductive probabilistic logic programming. In this paper we focus on latent feature modelling approaches, and refer the reader to [1] for a review of the other two. Techniques based on tensor factorisation have become popular in latent feature modelling based link prediction, and decompose relational data represented as a tensor of relational scores to generate a set of constituent vectors.

The RESCAL model by Nickel et al. [12] decomposes a relational score tensor S into an entity matrix $E \in \mathbb{R}^{n \times r}$, a relational tensor $R \in \mathbb{R}^{r \times r \times m}$ and the transpose of the entity matrix $E^T \in \mathbb{R}^{r \times n}$, as follows:

$$S \approx ERE^T. \tag{1}$$

The entity matrix E is composed of entity vectors $e_i \in \mathbb{R}^{1 \times r}$, and the relational tensor R is composed of full-rank matrix slices $W_k \in \mathbb{R}^{r \times r}$, where $k \in \{1, \ldots, m\}$ is the set of KG relations. The vector-matrix product between entity e_i and relational matrix W_k generates an entity-relational vector $h_{i,k} \in \mathbb{R}^{1 \times r}$. The inner product between $h_{i,k}$ and entity vector $e_j \in \mathbb{R}^{r \times 1}$ generates a relational score $s_{i,k,j}$ indicating the plausibility of entities i and j being linked by relation k. In this framework the entities and relations have latent vector representations computed using a parameterised function, and the idea would be for the model to learn optimal parameters from data. Training RESCAL amounts to minimising a squared error between known facts and model predictions.

The TransE model by Bordes et al. [19] follows a premise that many KG facts are presented in hierarchies, and that a translation of the subject vector by the relation should produce an embedding close to the object. The DistMul model by Yang et al. [20] first transforms the subject and object vectors into low-dimensional representations, and then applies a bilinear tensor product using a diagonal relation matrix. This approach effectively models a subset of the entity-relational interactions of RESCAL, and relies on its dimensionality reduction to generate sufficient semantic information. The ComplEx model by Trouillon et al. [14] represents entities with complex vectors, and relations with complex diagonal matrices, to allow the modelling of antisymmetric interactions (such as the predicate “starred in”, for example) where the subject and object are not interchangeable. The mentioned approaches all rely on linear tensor factorisation for latent feature modelling. They scale well with large datasets but are limited in their expressiveness.

The neural tensor network (NTN) model by Socher et al. [13] extends the bilinear tensor product in two ways. Firstly, it makes use of a recursive network to score a composition between two entities, and adds that score to the output of the bilinear tensor product to produce a relational score. Secondly, the computed relational score is passed through a squashing nonlinearity in order to generate a measure of confidence in a potential relationship between the two entities. These extensions enable higher levels of expressiveness and increase prediction performance significantly.

Hohenecker and Lukasiewicz [21] extended the NTN model by pre-computing an object representation as an aggregation of all facts in which the object plays a part, and called the new model a relational NTN. The HolE model by Nickel et al. [11] uses a circular correlation during relation prediction. This operation is performed between a subject vector e_i and an object vector e_j, by sliding the object over the subject, and an inner product with the predicate vector then produces a relational score. The ConvE model by Dettmers et al. [15] introduces the convolutional operator in entity-relational modelling. A subject vector $e_i \in \mathbb{R}^r$

and a predicate vector $w_k \in \mathbb{R}^r$ are reshaped individually to $\overline{e_i} \in \mathbb{R}^{r_w \times r_h}$ and $\overline{w_k} \in \mathbb{R}^{r_w \times r_h}$, and then vertically concatenated to create an entity-relational matrix $M \in \mathbb{R}^{r_w \times 2r_h}$. Convolutions are performed on M using a set of trainable filters, shared among all subject-predicate combinations. This creates a feature map that is flattened and passed through a fully-connected layer with a nonlinearity. Inner products are taken with all object vectors to create relational scores, which can be converted to probabilities with the sigmoid function. The convolutional operator increases expressiveness by modelling entity-relation feature interactions across the entire line of concatenation.

The HypER model by Balažević et al. [16] extends convolutional tensor factorisation by using a hypernetwork [17] to generate 2D relation-specific convolutional filters as predicate matrices. A 2D convolution is then taken between a subject vector and predicate matrix, the resulting feature map is reshaped and passed through a fully-connected layer with a nonlinearity, and an inner product leads to relational scores. The hypernetwork may introduce covariate shift [22], or a change in the distribution of inputs to a current layer, due to the simultaneous update of current layer weights and previous layer weights during training. Covariate shift slows down training and may degrade model performance during inference.

We aim to further improve link prediction performance of HypER on standard benchmark datasets, by optimising various aspects of training. To this end, we update the base NTN training algorithm with early stopping, Adam optimisation [23] and hyperparameter random search [24]. We then address the potential covariate shift introduced by hypernetworks, and also use pre-trained word vectors for initialising entity and relational vector embeddings.

3 Neural Tensor Factorisation

Tensor factorisation has shown promise in domains that are high-dimensional and sparse, and lends itself to efficient GPU computation. In the context of KGs, relational score tensors are decomposed into an entity matrix, a relational tensor and a transposed entity matrix. The entity matrix is composed of latent entity representations generated with trainable parameters. The relational tensor is composed of relational matrix slices, also generated with trainable parameters. In practice the bilinear tensor product between the entity matrix, the relational tensor and the transposed entity matrix leads to unnormalised relational scores for potential triples, from which a link can be predicted as the argmax over those scores.

Early tensor factorisation approaches focused on limiting the total number of model parameters in order to scale to large KGs, often at the expense of sufficiently complex entity-relational interaction modelling. In this section we expound on recent significant milestones in latent feature modelling approaches to link prediction that take advantage of the expressiveness of neural compositional models. We also discuss our proposed improvements.

3.1 Neural Tensor Networks

Socher et al. [13] introduced the use of recursive entity representations in the composition of the relational score. Recursive networks try to capture the rules for word combinations by constructing a composition tree between words, in a sequence of words. The NTN model takes advantage of these compositional rules by adding them to the bilinear tensor product as follows:

$$g_{i,k,j} = u_k^T \, f\Big(e_i^T W_k^{[1:m]} e_j + V_k \begin{bmatrix} e_i \\ e_j \end{bmatrix} + b_k\Big). \tag{2}$$

Here $g_{i,k,j}$ is the relational score tensor for subject i, object j and relation k, $u_k \in \mathbb{R}^{m\times 1}$ is an output layer of trainable weights, and f is the tanh activation function. $e_i^T W_k^{[1:m]} e_j$ is the bilinear tensor product between subject vector $e_i \in \mathbb{R}^{r\times 1}$ and object vector $e_j \in \mathbb{R}^{r\times 1}$, for relations $W_k \in \mathbb{R}^{r\times r}$, and b_k is a trainable bias vector. $V_k[\, e_i^T \;\; e_j^T \,]^T$ is the recursive entity composition, with $V_k \in \mathbb{R}^{m\times 2r}$ a matrix of trainable parameters.

During training a contrastive max-margin loss is minimised. This loss incorporates the score $g(T^{(n)})$ of a correct sample (a fact present in the KG) and the score $g(T_c^{(n)})$ of a corrupt sample (a randomly generated fact not present in the KG), as follows:

$$J(\Omega) = \sum_{n=1}^{N} \sum_{c=1}^{C} \max\big(0, 1 - g(T^{(n)}) + g(T_c^{(n)})\big) + \lambda ||\Omega||_2^2, \tag{3}$$

where Ω contains all the trainable parameters, N is the number of training samples, and C is the number of randomly corrupted facts to be used per true fact. The hyperparameter λ controls the importance of the ridge regulariser in this loss.

Doss et al. [18] reimplemented the NTN model in TensorFlow. This reimplementation underperforms compared to the original model (as we also find in Section 4.3), relying on AdaGrad optimisation and the same hyperparameters. In an attempt to improve the performance of the reimplementation we apply early stopping, the more modern Adam optimisation algorithm [23] as well as hyperparameter random search [24]. Early stopping tries to prevent overfitting, by tracking performance on the validation set during training and halting the training process as soon as it decreases significantly. Adam is a gradient based stochastic optimisation algorithm that tries to compensate for the sparse gradient signal problem, by incorporating first- and second-order moments of the gradient and parameter-specific adaptive learning rates. Hyperparameter random search defines intervals for all model hyperparameters and randomly samples from those intervals for a set number of training runs, thus taking advantage of the hypothesis that for many datasets only a subset of hyperparameters contribute meaningful variance in model performance, and eliminating the need for a costly exhaustive search.

3.2 Convolutional Networks

The ConvE model of Dettmers et al. [15] introduces the convolutional operator to neural tensor factorisation Specifically, this operator increases expressiveness in entity-relation interaction modelling by using 2D convolutions, which have been found to be particularly effective at modelling the interactions of entities involved in a large number of relations. This may be due to filter parameter sharing between entity-relational combination features, but perhaps more pertinently, the convolutional operator captures a larger variety of entity-relational feature interactions when summarising regions between the respective representations, whereas the bilinear tensor product performs interaction modelling using a simple inner product.

ConvE concatenates subject and predicate matrices along the row axis, and takes convolutions with trainable filters to produce feature maps. The feature maps are flattened and passed through a fully-connected layer with ReLU activation, to generate a latent vector representation. The inner product of this vector and the object vector then leads to an unnormalised relational score for the triple, which is passed through softmax normalisation.

During training a cross-entropy loss is minimised. This loss function is particularly appropriate as we expect only a single object to belong to the fact (subject-predicate-object) and generate a probability close to 1, and every other object to not belong to the fact, generating a probability close to 0.

3.3 Hypernetworks

The HypER model of Balažević et al. [16] extends convolutional tensor factorisation by using a hypernetwork [17] to generate 2D relation-specific convolutional filters as predicate matrices. A hypernetwork is a meta-network that generates parameters for a main network. It compresses those parameters into an input embedding vector, which is analogous to encoding a main network configuration and effectively reduces the total number of parameters without sacrificing performance. The embedding parameters of the hypernetwork are learned during end-to-end training of the main network.

HyPER makes use of a hypernetwork to generate relation-specific convolutional filters, where the subject and predicate filter are used in a convolution operation to generate a subject-predicate feature map. The feature map is flattened into a vector and passed through a fully-connected layer which outputs a hidden vector that is passed through a ReLU nonlinearity. An inner product is then taken with the object vector to produce an unnormalised relational score, and finally a softmax normalisation is applied to generate probabilities for potential relationships between pairs of entities.

We make the observation that hypernetworks may suffer from covariate shift. The hypernetwork generates relational filters for the main network from relational embedding inputs. During training, the distribution of the latent parameters of the relational embeddings could change, altering the inputs used to generate the filters. This change in distribution may make it hard for the hypernetwork's fully-connected layer to learn the most useful parameters for creating relational filters. Moreover, since the relational filters are used early (upstream) in the main network, the effect of their suboptimal weights might be amplified. To address this issue, we introduce relational input batch normalisation [22]. This operation performs normalisation on relational input batches during training, such that each batch has zero mean and unit variance, to regulate hypernetwork input and prevent covariate shift.

We refer to our improved model, which incorporates the training optimisations mentioned at the end of Section 3.1 as well as batch normalisation in the hypernetwork, as HypER+.

3.4 Pre-trained Word Vectors

As an additional potential improvement we also incorporate pre-trained word vectors from the GloVe language model [25]. Language models capture statistical correlations between words in a language, and in so doing attempt to build an understanding of the semantics of that language. Word meaning is embedded (or encoded) in a vector space, where words that share semantic meaning may be collocated. The GloVe model generates word vectors from both local and global co-occurrence statistics of words in a text corpus, and has been found to outperform other popular models (like Word2Vec).

KGs are comprised of vocabularies of entities and relations. These vocabularies contain a KG word to ID map, which is used during model training to identify the respective entity or relation. A pre-trained GloVe language model is a map of words to vectors, and can be used to look up the corresponding vectors for entities or relations in the KG.

Instead of the standard random initialisation of word vector embeddings, we experiment with using pre-trained GloVe word vectors to initialise 200-dimensional entity and relation embeddings. The respective embeddings are generated by aggregating the set of vectors corresponding to the sequence of words that describe them (where a pre-trained vector does not exist for a particular word, a randomly initialised vector can be generated in its place). This process generates two KG pre-trained vector to ID maps, one for entities and one for relations, and these maps are used to initialise the model entity and relation embeddings respectively. The embeddings are trainable, and updated during end-to-end training of HypER+ to generate representations specific to the KG under consideration.

4 Experiments

We proceed with an empirical examination of the three contributions of this work, namely (1) the application of training optimisation to neural tensor networks, (2) compensating for covariate shift in hypernetworks used for convolutional tensor factorisation, and (3) the initialisation of entity and relation embeddings with pre-trained word vectors.

We evaluate our models on three pairs of progressively more challenging benchmark link prediction datasets (detailed in Section 4.1), and compare a number of performance metrics (explained in Section 4.2) against existing and current state-of-the-art models (Section 4.3). The different datasets illuminate specific aspects, and informal experimentation with exhaustive training of all models on all datasets suggests trends similar to those reported in this paper.

All code needed to train and test our models, and reproduce the results in this section, are available at https://github.com/xhosaBoy.

4.1 Benchmark Datasets

Two of the earliest datasets used to evaluate link prediction are called WN11 and FB13, and were extracted from WordNet [26] and Freebase [27], respectively. WordNet is a lexical database for English, containing a taxonomy with hypernym relationships ("is a") and synonym sets. Freebase is a large collaborative knowledge base composed mainly by community members. WN11 contains just over 125,000 triples, split into training, validation and test sets as indicated in Table 1, with 38,696 unique entities (subjects and objects, each corresponding to a WordNet synset) and 11 unique relations (predicates). FB13 has close to 350,000 triples in total, with 75,043 unique entities and 13 relations.

The WN18 and FB15k datasets were introduced later by Bordes et al. [19]. WN18 consists of about 150,000 triples, split into training, validation and tests sets as indicated in Table 1, and includes 18 unique relations for about 41,000 unique entities from WordNet. FB15k has almost 600,000 triplets, consisting of 14,951 unique entities and 1,345 unique relations.

It was found that WN18 and FB15k both suffer from significant label leakage through inverse relations [15, 28], that is to say many triples in the test sets occur in inverted form in the training sets, making it easy for simple models to do well. Datasets WN18RR [15] and FB15k-237 [28] were created in an effort to avoid such leakage, through the removal of inverse relations from WN18 and FB15k, respectively. WN18RR has just over 93,000 triples over about 41,000 entities and 11 relations, while FB15k-237 has about 310,000 triples over 14,541 entities and 237 relations.

Table 1. The number of triples in the training, validation and tests sets of the three pairs of benchmark datasets considered in this study, along with the number of unique entities and unique relations in each.

Dataset	**Train**	**Val**	**Test**	**Entities**	**Relations**
WN11 [26]	112,581	2,609	10,544	38,696	11
FB13 [27]	316,232	5,908	23,733	75,043	13
WN18 [19]	141,442	5,000	5,000	40,943	18
FB15k [19]	483,142	50,000	59,071	14,951	1,345
WN18RR [15]	86,835	3,034	3,134	40,943	11
FB15k-237 [28]	272,115	17,535	20,466	14,541	237

4.2 Performance Metrics

We report a set of standard performance metrics used in the link prediction literature, namely Hit@10, Hit@3, Hit@1 and Mean Reciprocal Rank (MRR). Hit@k (or H@k for short) measures the fraction of times the correct label occurs in the top k predictions, if outputs are ordered by confidence scores. MRR is the average predicted inverse rank of correct labels. For example, if the true label is predicted as the third most likely output, the predicted inverse rank for that sample is 1/3.

All of these metrics will be expressed as percentages, and measured over the respective test sets of the various benchmark datasets, using only the final versions of models (after hyperparameter selection and training).

4.3 Results

For the first set of experiments we use the WN11 and FB13 datasets for training, validation and testing. We compare the test prediction accuracy (Hit@1) achieved by the original NTN model, as reported by Socher et al. [13], with the TensorFlow reimplementation [18] and our optimised version that incorporates early stopping, the Adam optimiser and hyperparamter random search.

Results are presented in Table 2. Our model outperforms the TensorFlow reimplementation, and quite significantly on the WN11 dataset. Training loss curves suggest that hyperparameter random search is mostly responsible for this improvement, as our model begins to outperform the baseline from the first training epoch. Reported results of the original NTN model are still far superior, possibly due to other (unknown) training algorithm modifications and optimisations.

Table 2. Link prediction accuracies (as percentages) achieved by the indicated models on the WN11 and FB13 test sets.

Model	**WN11**	**FB13**
Original NTN model [13]	86.2	90.0
TensorFlow reimplemented NTN [18]	56.2	53.5
Optimised (ours)	67.4	54.8

For the second set of experiments we use the WN18 and FB15k datasets for training, validation and testing. In Table 3 we compare the test set performance of our HypER+ model against HypER and also a suite of previous state-of-the-art models. Two models not mentioned in Section 2 are included here for completeness, namely TorusE [6] and R-GCN [29]. TorusE is an extension of the TransE model that circumvents a certain regularisation problem, while R-GCN models relational data with graph convolutional networks.

On the WN18 dataset our HypER+ model achieves results very close to the HypER model, across all metrics. The R-GCN model achieves the highest

Table 3. Link prediction results on the WN18 and FB15k test sets, as achieved by the indicated models. Our HypER+ model is the bottom row, and best results per column are shown in bold.

Model	**WN18**				**FB15k**			
	H@10	H@3	H@1	MRR	H@10	H@3	H@1	MRR
TransE [19]	89.2	-	-	-	47.1	-	-	-
DistMul [20]	93.6	91.4	72.8	82.2	82.4	73.3	54.6	65.4
ComplEx [14]	94.7	93.6	93.6	94.1	84.0	75.9	59.9	69.2
R-GCN [29]	**96.4**	92.9	69.7	81.9	84.2	76.0	60.1	69.6
TorusE [6]	95.4	95.0	94.3	94.7	83.2	77.1	67.4	73.3
ConvE [15]	95.6	94.6	93.5	94.3	83.1	72.3	55.8	65.7
HypER [16]	95.8	**95.5**	**94.7**	**95.1**	88.5	82.9	73.4	79.0
HypER+ (ours)	95.7	95.4	94.6	95.0	**89.4**	**85.6**	**79.0**	**82.9**

Hit@10 performance, but performs relatively poorly across the other metrics (especially Hit@1). On the FB15k dataset our HypER+ model outperforms all other models significantly. The impact of batch normalisation in the hypernetwork is pronounced, perhaps due to the upstream influence of predicate input covariate shift.

For the third and final set of experiments we use the WN18RR and FB15k-237 datasets for training, validation and testing. These datasets are more challenging subsets of the previous two (WN18 and FB15k), with inverse relations removed to prevent test leakage. Here we compare two versions of our HypER+ model with previous state-of-the-art models: one with randomly initialised embedding vectors and one initialised with pre-trained GloVe word vectors. Results are listed in Table 4.

Our HypER+ model with pre-trained GloVe initialisation leads to state-of-the-art performance across both WN18RR and FB15k-237. The pre-trained embeddings have a particularly pronounced impact on the Hit@10 metric over WN18RR, perhaps due to the smaller number samples in this dataset, but a diminished impact on the Hit@1 metric. Overall it seems as if the pre-trained

Table 4. Link prediction results on the more challenging WN18RR and FB15k-237 test sets, as achieved by the indicated models. Our original HypER+ model and one that was initialised at training time with pre-trained GloVe word vectors form the last two rows, and best results per column are shown in bold.

Model	**WN18RR**				**FB15k-237**			
	H@10	H@3	H@1	MRR	H@10	H@3	H@1	MRR
DistMul [20]	49.0	44.0	39.0	43.0	41.9	26.3	15.5	24.1
ComplEx [14]	51.0	46.0	41.0	44.0	42.8	27.5	15.8	24.7
ConvE [15]	52.0	44.0	40.0	43.0	50.1	35.6	23.7	32.5
HypER [16]	52.2	47.7	43.6	46.5	52.0	37.6	25.2	34.1
HypER+ (ours)	51.9	47.9	**43.8**	46.6	51.6	36.8	24.5	33.5
GloVe init (ours)	**57.8**	**49.3**	43.5	**48.0**	**52.5**	**37.9**	**25.5**	**34.5**

embeddings may provide a more meaningful contribution than batch normalisation in the hypernetwork, as suggested by the inferior performance of HypER+ compared to HypER. Somewhat strangely, this is inconsistent with training and validation accuracies, where HypER+ consistently outperformed HypER. The fact that we use the quoted HypER test results, as opposed to reimplemented and verified test results, may be a partial explanation. That inconsistency aside, pre-trained embeddings from the GloVe language model appear to be effective at improving link prediction performance.

Figure 1 shows the Hit@1 test set accuracy for each of 10 relations from WN18RR, as achieved by our HypER+ model initialised with pre-trained word vectors. The model performs well with synonym relation types, but poorly with compositional and hierarchical relations. This may be due to the inherent similarity and analogy in concepts, whereas compositions and hierarchies can be defined by strict rules. Perhaps, more simply, it could be due to the number of test set samples for each relation, where higher numbers increase the prediction error rate.

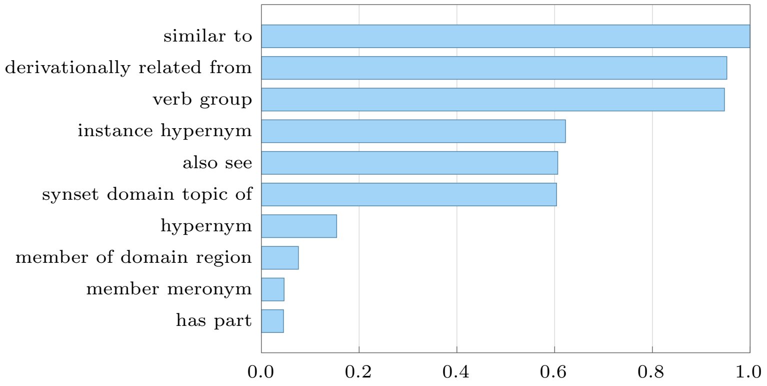


Fig. 1. Hit@1 prediction accuracy per predicate, as achieved by the HypER+ model with pre-trained GloVe word vectors on the WN18RR test set. It should be noted that the predicates are not evenly distributed in the dataset; “hypernym” and “derivationally related from” respectively account for about 40% and 35% of all triples in WN18RR.

For illustrative purposes we show in Table 5 a handful of incorrect test set predictions from our HypER+ model with pre-trained word vectors. These examples are all for the predicate “has part”, which according to Figure 1 seems difficult to model accurately. Our model does demonstrate basic conceptual understanding, as most of these mistakes could be deemed forgivable. We would therefore expect reasonable knowledge discovery utility from the model, when used jointly with information retrieval for open-domain question answering.

Table 5. Examples of incorrect predictions made by our HypER+ model with pre-trained word vectors on the WN18RR test set.

Subject	Predicate	Target object	Predicted object
usa	has part	colorado	missouri river
spain	has part	cadiz	jerez de la frontera
electromagnetic spectrum	has part	actinic ray	radio spectrum
systema respiratorium	has part	respiratory tract	respiratory organ
africa	has part	nigeria	senegal
antigen	has part	substance	epitope
amphitheatre	has part	theatre	tiered seat
indian ocean	has part	mauritius	antarctic ocean

5 Conclusions

Neural tensor factorisation has shown promise in extending the performance of latent feature modelling based link prediction in knowledge graphs. The original insight to apply tensor decompositions to relational modelling gave rise to a number of increasingly performant models, and ultimately to our proposed HypER+ model.

There remains an open question as to the extent to which representation structure can continue to improve performance. HypER [16] for example extends the concept of multi-dimensional inputs proposed by ConvE [15], by generating relational matrices as opposed to relying on relational vectors. A natural further extension is generating entity matrices, and then potentially tensor representations for both. Such dense representations may better capture entity and relational concepts. The application of alternative entity-relation feature interaction operators is another natural extension. It has been demonstrated [14, 11, 15] that the Hermitian dot product, circular correlation and convolution operators have potential. An as yet unexplored avenue is the use of stochastic operators that directly encode uncertainty, and may produce relational score probabilities with more appropriate confidence measures.

Some progress in link prediction was recently achieved using the graph modelling paradigm. State-of-the-art Hit@1 accuracy of 46% was achieved on FB15k-237 by Nathani et al. [30], compared to an accuracy of 25.5% achieved by our HypER+ model with pre-trained word vectors. Such a staggering improvement is however not realised on WN18RR, with the model of Nathani et al. achieving a Hit@1 accuracy of 36.1% compared to our model's 43.5%. But these graph modelling approaches clearly demonstrate potential in pushing link prediction performance forward. Their strength could be exploiting a similar idea to the one used to construct the GloVe language model, incorporating both local and global context. Inductive probabilistic logic programming (IPLP) also shows promise [31, 32], although research in this paradigm is much more sparse. It would seem graph modelling, as opposed to latent feature modelling, may provide the biggest contribution to link prediction in the near-term.

References

1. Nickel, M., Murphy, K., Tresp, V., Gabrilovich, E.: A review of relational machine learning for knowledge graphs. Proceedings of the IEEE **104**(1), 11–33 (2016)
2. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia - a crystallization point for the web of data. Journal of Web Semantics **7**(3), 154–165 (2009)
3. Getoor, L., Taskar, B. (eds.): Introduction to Statistical Relational Learning. MIT Press (2007)
4. Chen, D., Fisch, A., Weston, J., Bordes, A: Reading Wikipedia to answer open-domain questions. In: Meeting of the Association for Computational Linguistics, pp. 1870–1879 (2017)
5. Kristiadi, A., Khan, M., Lukovnikov, D., Lehmann, J., Fischer, A.: Incorporating literals into knowledge graph embeddings. In: International Semantic Web Conference, pp. 347–363 (2019)
6. Ebisu, T., Ichise, R.: TorusE: knowledge graph embedding on a Lie group. In: AAAI Conference on Artificial Intelligence, pp. 1819–1826 (2018)
7. Nguyen, D., Nguyen, T., Nguyen, D., Phung, D.: A novel embedding model for knowledge base completion based on convolutional neural network. In: Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 327–333 (2018)
8. Kolda, T., Bader, B.: Tensor decompositions and applications. SIAM Review **51**(3), 455–500 (2009)
9. Bordes, A., Weston, J., Collobert, R., Bengio, Y.: Learning structured embeddings of knowledge bases. In: AAAI Conference on Artificial Intelligence, pp. 301–306 (2011)
10. Jenatton, R., Roux, N., Bordes, A., Obozinski, G.: A latent factor model for highly multi-relational data. In: Advances in Neural Information Processing Systems, pp. 3167–3175 (2012)
11. Nickel, M., Rosasco, L., Poggio, T.: Holographic embeddings of knowledge graphs. In: AAAI Conference on Artificial Intelligence, pp. 1955–1961 (2016)
12. Nickel, M., Tresp, V., Kriegel, H.: A three-way model for collective learning on multi-relational data. In: International Conference on Machine Learning, pp. 809–816 (2011)
13. Socher, R., Chen, D., Manning, C., Ng, A.: Reasoning with neural tensor networks for knowledge base completion. In: Advances in Neural Information Processing Systems, pp. 926–934 (2013)
14. Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., Bouchard, G.: Complex embeddings for simple link prediction. In: International Conference on Machine Learning, pp. 2071–2080 (2016)
15. Dettmers, T., Minervini, P., Stenetorp, P., Riedel, S.: Convolutional 2D knowledge graph embeddings. In: AAAI Conference on Artificial Intelligence, pp. 1811–1818 (2018)
16. Balažević, I., Allen, C., Hospedales, T.: Hypernetwork knowledge graph embeddings. In: International Conference on Artificial Neural Networks, pp. 553–565 (2019)
17. Ha, D., Dai, A., Le, Q.: Hypernetworks. In: International Conference on Learning Representations (2017)
18. Doss, D., LeNail, A., Liu, C. (2015). Reimplementing neural tensor networks for knowledge base completion in the TensorFlow framework. https://github.com/

dddoss/tensorflow-socher-ntn/blob/master/notes/paper.pdf. Last accessed 15 Sep 2020

19. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Advances in Neural Information Processing Systems, pp. 2787–2795 (2013)
20. Yang, B., Yih, S., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases. In: International Conference on Learning Representations (2015)
21. Hohenecker, P., Lukasiewicz, T.: Ontology reasoning with deep neural networks. Journal of Artificial Intelligence Research **68**, 503–540 (2020)
22. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate Shift. In: International Conference on Machine Learning, pp. 448–456 (2015)
23. Kingma, D., Ba, J.: Adam: a method for stochastic optimization. In: International Conference on Learning Representations (2015)
24. Bergstra, J., Bengio, Y.: Random search for hyper-parameter optimization. Journal of Machine Learning Research **13**(Feb), 281–305 (2012)
25. Pennington, J., Socher, R., Manning, C.: GloVe: global vectors for word representation. In: Conference on Empirical Methods in Natural Language Processing, pp. 1532–1543 (2014)
26. Miller, G.: WordNet: a lexical database for English. Communications of the ACM **38**(11), 39–41 (1995)
27. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: a collaboratively created graph database for structuring human knowledge. In: ACM International Conference on Management of Data, pp. 1247–1250 (2008)
28. Toutanova, K., Chen, D.: Observed versus latent features for knowledge base and text inference. In: Workshop on Continuous Vector Space Models and their Compositionality, pp. 57–66 (2015)
29. Schlichtkrull, M., Kipf, T., Bloem, P., Van Den Berg, R., Titov, I., Welling, M.: Modeling relational data with graph convolutional networks. In: European Semantic Web Conference, pp. 593–607 (2018)
30. Nathani, D., Chauhan, J., Sharma, C., Kaul, M.: Learning attention-based embeddings for relation prediction in knowledge graphs. In: Meeting of the Association for Computational Linguistics, pp. 4710–4723 (2019)
31. Dong, X., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., Strohmann, T., Sun, S., Zhang, W.: Knowledge Vault: a web-scale approach to probabilistic knowledge fusion. In: ACM International Conference on Knowledge Discovery and Data Mining, pp. 601–610 (2014)
32. Manhaeve, R., Dumancic, S., Kimmig, A., Demeester, T., De Raedt, L.: DeepProbLog: neural probabilistic logic programming. In: Advances in Neural Information Processing Systems, pp. 3749–3759 (2018)

Author Index