Zbornik 24. mednarodne multikonference

INFORMACIJSKA DRUŽBA – IS 2021

Zvezek C

Proceedings of the 24th International Multiconference

INFORMATION SOCIETY – IS 2021

Volume C

Odkrivanje znanja in podatkovna skladišča - SiKDD
Data Mining and Data Warehouses - SiKDD

Urednika / Editors

Dunja Mladenić, Marko Grobelnik

http://is.ijs.si

4. oktober 2021 / 4 October 2021
Ljubljana, Slovenia

Urednika:

Dunja Mladenić,
Department for Artificial Intelligence
Jožef Stefan Institute, Ljubljana

Marko Grobelnik
Department for Artificial Intelligence
Jožef Stefan Institute, Ljubljana

Založnik: Institut »Jožef Stefan«, Ljubljana
Priprava zbornika: Mitja Lasič, Vesna Lasič, Lana [illegible]ljak
Oblikovanje naslovnice: Vesna Lasič

Dostop do e-publikacije:
http://library.ijs.si/Stacks/Proceedings/InformationSociety

Ljubljana, oktober 2021

Informacijska družba
ISSN 2630-371X

PREDGOVOR MULTIKONFERENCI INFORMACIJSKA DRUŽBA 2021

Štiriindvajseta multikonferenca Informacijska družba (http://is.ijs.si) je preživela probleme zaradi korone v 2020. Odziv se povečuje, v 2021 imamo enajst konferenc, a pravo upanje je za 2022, ko naj bi dovolj velika precepljenost končno omogočila normalno delovanje. Tudi v 2021 gre zahvala za skoraj normalno delovanje konference tistim predsednikom konferenc, ki so kljub prvi pandemiji modernega sveta pogumno obdržali visok strokovni nivo.

Stagnacija določenih aktivnosti v 2020 in 2021 pa skoraj v ničemer ni omejila neverjetne rasti IKT, informacijske družbe, umetne inteligence in znanosti nasploh v 2021, ampak nasprotno, rast znanja, računalništva in umetne inteligence se nadaljuje z že kar običajno nesluteno hitrostjo. Po drugi strani pa se je še dodatno pospešil razpad družbenih vrednot, zaupanje v znanost in razvoj, kar se kaže predvsem v raznih protivladnih gibanjih. Žal čedalje več ljudi verjame, da je Zemlja ploščata, da je cepivo za korono škodljivo, da virusov ni … Razkorak med rastočim znanjem in vraževerjem se povečuje tudi v zadnjem letu, zavedanje večine ljudi, da to vodi nazaj v srednji vek, čedalje bolj krepi, kar je bistvena sprememba glede na 20…

Letos smo v multikonferenco povezali enajst odličnih neodvisnih konferenc. … okoli 150 večinoma spletnih predstavitev, povzetkov in referatov v okviru samostojnih konferenc in delavnic … obiskovalcev. Prireditev so spremljale okrogle mize in razprave ter posebni dogodki, kot je svečana podelitev nagrad – seveda večinoma preko spleta. Izbrani prispevki bodo izšli tudi v posebni številki revije Informatica (http://www.informatica.si/), ki se ponaša s 45-letno tradicijo odlične znanstvene revije.

Multikonferenco Informacijska družba 2021 sestavljajo naslednje samostojne konference:

- Slovenska konferenca o umetni inteligenci
- Odkrivanje znanja in podatkovna skladišča
- Kognitivna znanost
- Ljudje in okolje
- 50-letnica poučevanja računalništva v slovenskih srednjih šolah
- Delavnica projekta Batman
- Delavnica projekta Insieme Interreg
- Delavnica projekta Urbanite
- Študentska konferenca o računalniškem raziskovanju 2021
- Mednarodna konferenca o prenosu tehnologij
- Vzgoja in izobraževanje v informacijski družbi

Soorganizatorji in podporniki konference so različne raziskovalne institucije in združenja, med njimi ACM Slovenija, SLAIS, DKZ in druga slovenska nacionalna akademija, Inženirska akademija Slovenije (IAS). V imenu organizatorjev konference se zahvaljujemo združenjem in institucijam, še posebej pa udeležencem za njihove dragocene prispevke in priložnost, da z nami delijo svoje izkušnje o informacijski družbi. Zahvaljujemo se tudi recenzentom za njihovo pomoč pri recenziranju.

Nagrade bodo proglašene v petek, 8.10.2021. Podelili bomo nagrado za življenjske dosežke v čast Donalda Michieja in Alana Turinga. Nagrado Michie-Turing za izjemen življenjski prispevek k razvoju in promociji informacijske družbe je prejme …. Priznanje za dosežek leta pripada …. Podeljujemo tudi nagradi »informacijska limona« in »informacijska jagoda« za najbolj (ne)uspešne poteze v zvezi z informacijsko družbo. Limono prejme …, jagodo …. Čestitke nagrajencem!

Mojca Ciglarič, predsednik programskega odbora
Matjaž Gams, predsednik organizacijskega odbora

FOREWORD - INFORMATION SOCIETY 2021

The 24th Information Society Multiconference (http://is.ijs.si) survived the COVID-19 problems. In 2021, there are eleven conferences with a growing trend and real hopes that 2022 will be [illegible] due to successful vaccination. The multiconference survived due to the conference presidents that bravely decided to continue with their conference despite the first pandemics in the modern era.

The COVID-19 pandemic did not decrease the growth of ICT, information society, artificial intelligence and science overall, quite on the contrary – the progress of computers, knowledge and artificial intelligence continued with the fascinating growth rate. However, COVID-19 did increase the downfall of societal trust in science and progress, most evident in anti-vaccination movements. The number of people believing that the Earth is flat is growing as well as those that believe that the COVID-19 vaccines are harmful or even that viruses don't exist at all. On the other hand, the awareness of the majority population that such approaches could lead to returning to the Dark Ages, grows to the point that proper actions against this phenomenon are promoted.

The Multiconference is running parallel sessions with 150 presentations of scientific papers at eleven conferences, many round tables, workshops and award ceremonies and 300 attendees. Selected papers will be published in the Informatica journal with its 45-years tradition of excellent research publishing.

The Information Society 2021 Multiconference consists of the following conferences:

- Slovenian Conference on Artificial Intelligence
- Data Mining and Data Warehouses
- Cognitive Science
- People and Environment
- 50-years of High-school Computer Education in Slovenia
- Batman Project Workshop
- Insieme Interreg Project Workshop
- URBANITE Project Workshop
- Student Computer Science Research Conference 2021
- International Conference of Transfer of Technology
- Education in Information Society

The multiconference is co-organized and supported by several major research institutions and societies, among them ACM Slovenia, i.e. the Slovenian chapter of the ACM, SLAIS, GZ and the second national engineering academy, the Slovenian Engineering Academy. In the name of the conference organizers, we thank all the societies and institutions, and particularly all the participants for their valuable contribution and their interest in this event, and the reviewers for their thoughtful views.

The awards will be announced on 8.10.2021. The award for life-long outstanding contributions will be presented in memory of Donald Michie and Alan Turing. The Michie-Turing award was given to … for his life-long outstanding contribution to the development and promotion of information society in our country. In addition, a recognition for current achievements was awarded to … The information lemon goes to the …, and the information strawberry to the … Congratulations!

Mojca Ciglarič, Programme Committee Chair
Matjaž Gams, Organizing Committee Chair

KONFERENČNI ODBORI
CONFERENCE COMMITTEES

International Programme Committee

Vladimir Bajic, Južna Afrika
Heiner Benking, Nemčija
Se Woo Cheon, Južna Koreja
Howie Firth, Škotska
Olga Fomichova, Rusija
Vladimir Fomichov, Rusija
Vesna Hljuz Dobric, Hrvaška
Alfred Inselberg, Izrael
Jay Liebowitz, ZDA
Huan Liu, Singapur
Henz Martin, Nemčija
Marcin Paprzycki, ZDA
Claude Sammut, Avstralija
Jiri Wiedermann, Češka
Xindong Wu, ZDA
Yiming Ye, ZDA
Ning Zhong, ZDA
Wray Buntine, Avstralija
Bezalel Gavish, ZDA
Gal A. Kaminka, Izrael
Mike Bain, Avstralija
Michela Milano, Italija
Derong Liu, Chicago, ZDA
Toby Walsh, Avstralija

Organizing Committee

Matjaž Gams, chair
Mitja Luštrek
Lana Zemljak
Vesna Koricki
Mitja Lasič
Blaž Mahnič
Jani Bizjak
Tine Kolenik

Programme Committee

Mojca Ciglarič, chair
Bojan Orel, co-chair
Franc Solina,
Viljan Mahnič,
Cene Bavec,
Tomaž Kalin,
Jozsef Györkös,
Tadej Bajd
Jaroslav Berce
Mojca Bernik
Marko Bohanec
Ivan Bratko
Andrej Brodnik
Dušan Caf
Saša Divjak
Tomaž Erjavec
Bogdan Filipič
Andrej Gams
Matjaž Gams
Mitja Luštrek
Marko Grobelnik
Nikola Guid
Marjan Heričko
Borka Jerman Blažič Džonova
Gorazd Kandus
Urban Kordeš
Marjan Krisper
Andrej Kuščer
Jadran Lenarčič
Borut Likar
Janez Malačič
Olga Markič
Dunja Mladenič
Franc Novak
Vladislav Rajkovič
Grega Repovš
Ivan Rozman
Niko Schlamberger
Stanko Strmčnik
Jurij Šilc
Jurij Tasič
Denis Trček
Andrej Ule
Tanja Urbančič
Boštjan Vilfan
Baldomir Zajc
Blaž Zupan
Boris Žemva
Leon Žlajpah
Niko Zimic
Rok Piltaver
Toma Strle
Tine Kolenik
Franci Pivec
Uroš Rajkovič
Borut Batagelj
Tomaž Ogrin
Aleš Ude
Bojan Blažica
Matjaž Kljun
Robert Blatnik
Erik Dovgan
Anton Gradišek
Lidija Zadnik Stirn
Marjan Mernik
Tomaž Pisanski
Janez Grad
Dušan Kodek
Vladimir Batagelj
Anton P. Železnikar

KAZALO / TABLE OF CONTENTS

Zbornik 24. mednarodne multikonference

INFORMACIJSKA DRUŽBA - IS 2021

Zvezek C

Proceedings of the 24th International Multiconference

INFORMATION SOCIETY – IS 2021

Volume C

Odkrivanje znanja in podatkovna skladišča - SiKDD
Data Mining and Data Warehouses - SiKDD

Urednika / Editors

Dunja Mladenić, Marko Grobelnik

http://is.ijs.si

Oktober 2021 / 4 October 2021
Ljubljana, Slovenia

PREDGOVOR

Tehnologije, ki se ukvarjajo s podatki so v devetdesetih letih močno napredovale. Iz prve faze, kjer je šlo predvsem za shranjevanje podatkov in kako do njih učinkovito dostopati, se je razvila industrija za izdelavo orodij za delo s podatkovnimi bazami, prišlo je do standardizacije procesov, povpraševalnih jezikov itd. Ko shranjevanje podatkov ni bil več poseben problem, se je pojavila potreba po bolj urejenih podatkovnih bazah, ki bi služile ne le transakcijskem procesiranju ampak tudi analitskim vpogledom v podatke – pojavilo se je t.i. skladiščenje podatkov (data warehousing), ki je postalo standarden del informacijskih sistemov v podjetjih. Paradigma OLAP (On-Line-Analytical-Processing) zahteva od uporabnika, da še vedno sam postavlja sistemu vprašanja in dobiva nanje odgovore in na vizualen način preverja in išče izstopajoče situacije. Ker seveda to ni vedno mogoče, se je pojavila potreba po avtomatski analizi podatkov oz. z drugimi besedami to, da sistem sam pove, kaj bi utegnilo biti zanimivo za uporabnika – to prinašajo tehnike odkrivanja znanja v podatkih (data mining), ki iz obstoječih podatkov skušajo pridobiti novo znanje in tako uporabniku nudijo novo razumevanje dogajanj zajetih v podatkih. Slovenska KDD konferenca pokriva vsebine, ki se ukvarjajo z analizo podatkov in odkrivanjem znanja v podatkih: pristope, orodja, probleme in rešitve.

FOREWORD

Data driven technologies have significantly progressed after mid 90’s. The first phases were mainly focused on storing and efficiently accessing the data, resulted in the development of industry tools for managing large databases, related standards, supporting querying languages, etc. After the initial period, when the data storage was not a primary problem anymore, the development progressed towards analytical functionalities on how to extract added value from the data; i.e., databases started supporting not only transactions but also analytical processing of the data. At this point, data warehousing with On-Line-Analytical-Processing entered as a usual part of a company’s information system portfolio, requiring from the user to set well defined questions about the aggregated views to the data. Data Mining is a technology developed after year 2000, offering automatic data analysis trying to obtain new discoveries from the existing data and enabling a user new insights in the data. In this respect, the Slovenian KDD conference (SiKDD) covers a broad area including Statistical Data Analysis, Data, Text and Multimedia Mining, Semantic Technologies, Link Detection and Link Analysis, Social Network Analysis, Data Warehouses.

PROGRAMSKI ODBOR / PROGRAMME COMMITTEE

Jane Brank, Jožef Stefan Institute, Ljubljana

Marko Grobelnik, Jožef Stefan Institute, Ljubljana

Jakob Jelenčič, Jožef Stefan Institute, Ljubljana

Branko Kavšek, University of Primorska, Koper

Aljaž Košmerlj, Qlector, Ljubljana

Dunja Mladenić, Jožef Stefan Institute, Ljubljana

Inna Novalija, Jožef Stefan Institute, Ljubljana

Jože Rožanec, Qlector, Ljubljana

Luka Stopar, Sportradar, Ljubljana

OBSERVING ODOR-RELATED INFORMATION IN ACADEMIC DOMAIN

Inna Novalija
Jožef Stefan Institute
Jamova cesta 39, Ljubljana, Slovenia
inna.koval@ijs.si

Dunja Mladenić
Jožef Stefan Institute and Jožef Stefan International Postgraduate School
Jamova cesta 39, Ljubljana, Slovenia
dunja.mladenic@ijs.si

Daniel Schwabe
Jožef Stefan Institute
Jamova cesta 39, Ljubljana, Slovenia
daniel.schwabe@ijs.si

M. Besher Massri
Jožef Stefan Institute and Jožef Stefan International Postgraduate School
Jamova cesta 39, Ljubljana, Slovenia
besher.massri@ijs.si

Marko Grobelnik
Jožef Stefan Institute
Jamova cesta 39, Ljubljana, Slovenia
marko.grobelnik@ijs.si

Janez Brank
Jožef Stefan Institute
Jamova cesta 39, Ljubljana, Slovenia
janez.brank@ijs.si

ABSTRACT

In this paper, we demonstrate an approach for observing olfactory related information in an academic publications environment (such as Microsoft Academic Graph) based on semantic technologies. We present an Odor Observatory tool that enables several usage scenarios, such as observing odor-related papers and topics, viewing institutions conducting olfactory research, defining top journals and key countries in the olfactory domain.

Validation of the proposed approach on a collection of academic publications from 1800 until 1925 confirms applicability of the proposed approach on large data collections with a wide span of time. In usage scenarios we observed the odor-related publications in Microsoft Academic Graph by topic, discovered the journals with historical olfactory publications and found that the most popular terms in odor-related research content are: method, olfactory, odor, device, invention, smell, preparation, utility model.

KEYWORDS

Odor, Olfactory information, Microsoft Academic Graph (MAG), Data mining.

1. INTRODUCTION

Olfaction, or the sense of smell, is the sense through which smells (or odors) are perceived [1]. Olfactory science involves studying olfaction and odor-related topics, the sensory system, physiology, and pheromone signals.

The Odeuropa project [2] gathers and integrates expertise in sensory mining and olfactory heritage. The project partners are developing novel methods to collect information about smell from (digital) text and image collections.
The Odeuropa project partners apply state-of-the-art AI techniques to text and image datasets in order to identify and trace how 'smell' was expressed in different languages, with what places it was associated, what kinds of events and practices it characterized, and to what emotions it was linked.

In this paper we present an approach for mining olfactory information from scientific research collections, such as the Microsoft Academic Graph (MAG) [3].

The olfactory mining approach combines data processing, modelling and visualization methods in order to develop applicable tools for data analysis.

We present an Odor Observatory tool [4] targeted at several visualization scenarios. In particular, the Odor Observatory allows exploring olfactory related papers from the MAG over time, and along with current data, provides historical information starting with the early XIX century.

The data-driven functionalities of Odor Observatory are:

- Possibility of exploring top ranked topics in the olfactory academic domain;
- Possibility of exploring top ranked institutions conducting olfactory research;
- Possibility of exploring key countries and defining top ranking journals in the olfactory academic domain;
- Odor-related search functionalities;
- Word cloud visualization for odor-related terms.

2. RELATED WORK

Olfactory science covers different aspects of research related to odors, therefore exploring odor related information and data can be viewed as complex multidisciplinary area.

Lötsch et al. [5] considered machine learning approaches for human olfactory research. The authors state that the complexity of the human sense of smell is reflected in complex and high-dimensional data, which supports the applicability of machine learning and data mining techniques. The use of machine learning in human olfactory research includes the following aims:

1. The study of the physiology of pattern-based odor detection and recognition processes;
2. Pattern recognition in olfactory phenotypes;
3. The development of complex disease biomarkers including olfactory features;

4. Odor prediction from physico-chemical properties of volatile molecules, and
5. Knowledge discovery in publicly available large databases.

The authors provide review of key concepts of machine learning and summarizes current applications on human olfactory data.

At the same time, linguistic and semantic communities focused on studying the language of smell [6]. Iatropoulos et al. developed a computational method to characterize the olfaction-related semantic content of words in a large text corpus of internet sites in English. They also introduced novel metrics, such as olfactory association index (OAI) and olfactory specificity index (OSI).

Tonelli [7] describes olfactory information extraction and semantic processing from a multilingual perspective. The author states that in several studies it was found that languages seem to have a smaller vocabulary to describe smells as compared to other senses.

In our work we apply data mining and machine learning, as well as semantic approaches for enriching textual data. We use data from Microsoft Academic Graph and our methodologies can be regarded as being in the context of semantic and text processing research. Our approaches can cover cross-lingual and multilingual data and allow for tracking olfactory trends in time.

3. PROBLEM DEFINITION

3.1 DATA SOURCES

The Microsoft Academic Graph (MAG) [3] is a heterogeneous graph containing scientific publication records, citation relationships between those publications, as well as authors, institutions, journals, conferences, and fields of study.

Since this research is conducted in line with the Odeuropa project (targeted at olfactory heritage), the time frame used for MAG data is set to range from the early publications in the 19th century to the present time. The Odeuropa project is interested in particular in the data available up to 1925. Though the project is focused on the historical datasets, the developed Odor Observatory tool allows users to explore recent olfactory publications as well. The dataset is updated on a monthly basis and new available data is uploaded into the observatory.

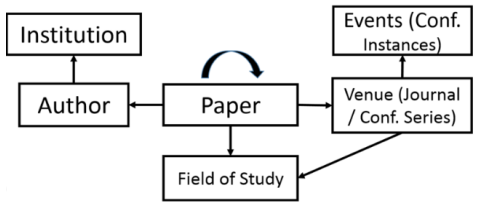

Figure 1: The Conceptual Schema for MAG

The Microsoft Academic Graph data schema is based on the list of following entity types: publication, author, author affiliation (institution), publication venue (journals and conferences), field of study (topic). It contains information about publication dates, as well as citation pairs and co-authorship data (see Figure 1).

Figure 2 illustrates an entry in MAG for a historical publication tagged with several odor-relevant topics.

Figure 2 illustrates an entry in MAG for a historical publication tagged with several odor-relevant topics.

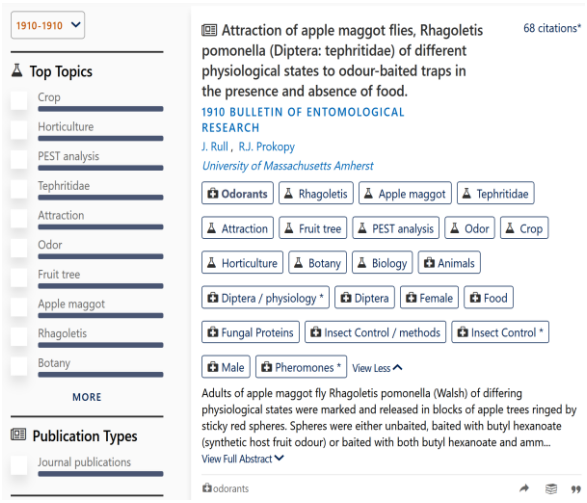

Figure 2: Publication in MAG

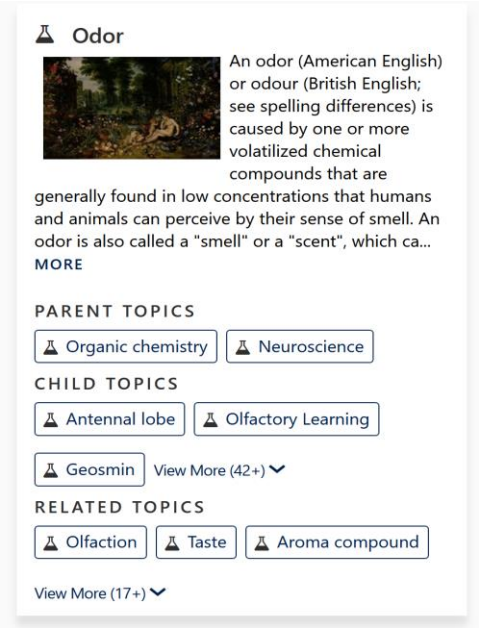

Figure 3: Odor in the MAG Taxonomy

Figure 3 shows a representation of Odor in the MAG taxonomy, with parent topics (Organic chemistry and Neuroscience) and child topics (Olfactory learning, Geosmin etc.)

An important functionality while exploring the literature is the ability to expand searches by looking at related topics to a topic of interest. Figure 4 displays topics about/related to Olfaction/Odor/Smell in MAG taxonomy.

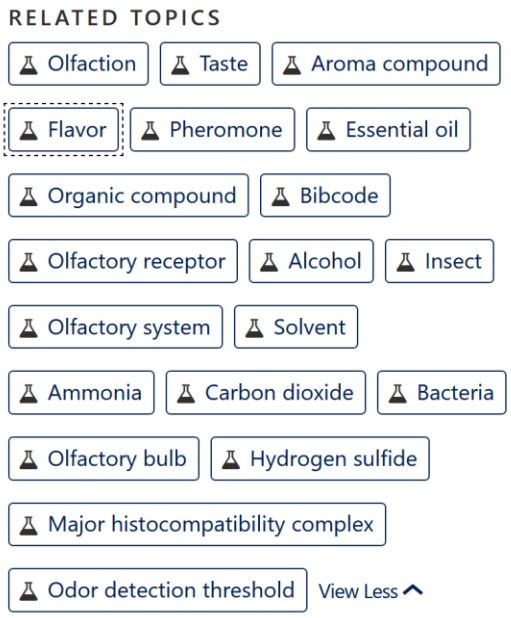

Figure 4: Odor-related Topics in MAG

3.2 METHODOLOGY

The methodology for observing olfactory related information from academic publication resources includes a number of steps:

- Using the MAG taxonomy, obtain the list of research papers that corresponds to odor-related topics. Papers were filtered to those containing the topics: Olfaction, Odor, Fragrance, Fragrance ingredient, as well as the "smell" keyword;
- Ingest the extracted corpus into the Elastic Search tool[1];
- Provide visualization functionalities, such as MAG time series per term.

The key challenges of the development techniques include:

- Interpretability and explainability of the results – the aim is for the visualizations to be able easily interpretable by humans;
- Given the large scale of the incoming data streams, it is essential that building visualizations are scalable. The MAG contains more than 265 million records (August 2021), including several types of publication, such as journal articles, conference papers, books, book chapters, and papers from other repositories. In addition, MAG also indexes a large corpus of patents.

3.3 USAGE SCENARIOS

We present a couple of usage scenarios for the Odor Observatory tool, cast as questions asked by scholars studying the field.

1. What are the historical trends in odor-related publications?

[1] https://www.elastic.co

Figure 5 shows the number of odor-related historical publications in MAG over time. This scenario assumes observing trends in different olfactory topics throughout a time interval.

It is possible to observe that the highest number of publications are in the domains of biology and psychology.

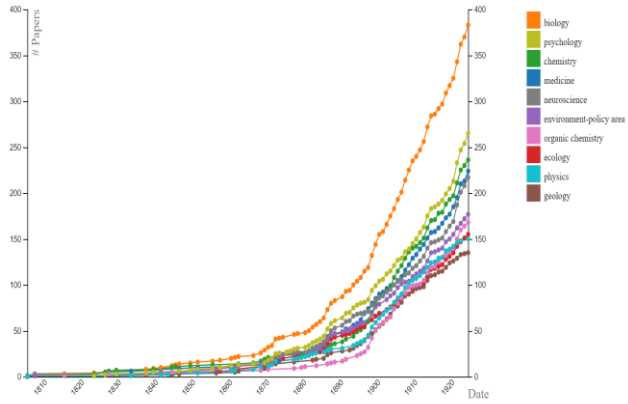

Figure 5: Odor-related Publications in MAG (from year 1800 until year 1925, cumulative) by topic

2. What are the most popular terms used in odor-related publications?

This use case helps the user to visualize term usage by displaying a word cloud with the most popular olfactory terms used in the publications in the period of interest (see Figure 6).

Figure 6: Odor Terms Word Cloud in MAG

3. Which venues were mostly used when publishing odor-related research articles?

This use case shows a number of journals that had historical publications about smells (see Figure 7).

The figure shows that JAMA and Nature journals are the most popular journals regarding historical olfactory publications.

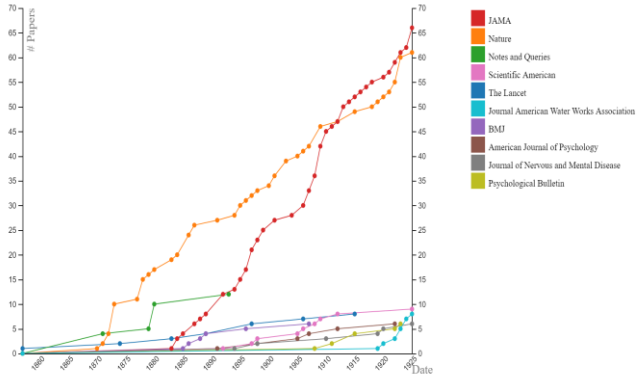

Figure 7: Journals with Olfactory Publications in MAG (from year 1800 until year 1925, cumulative)

4. Which are the publications about smell (from a contextual point of view)?

The Research Explorer tool is a search engine that enables exploring the individual articles in the corpus of odor-related publications.

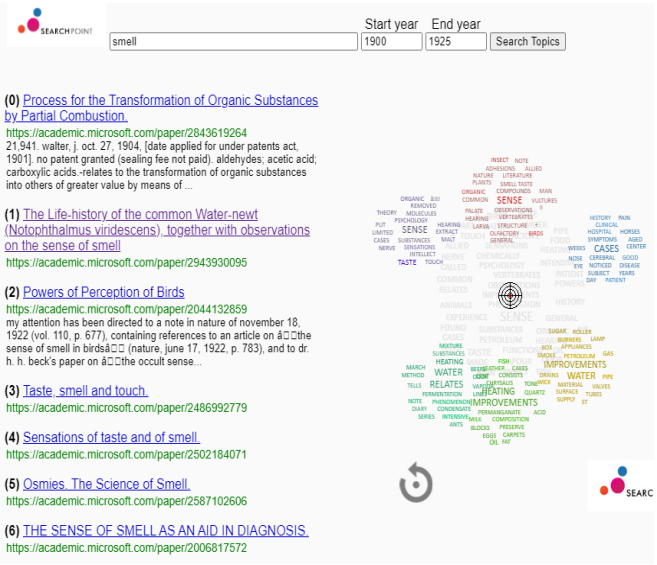

Figure 8: List of Olfactory Publications in MAG (from year 1800 until year 1925) that contains the keyword "smell"

The tool is built on Elastic Search and provides search by keyword and by date. It also supports smart navigation through the results by clustering the results and re-ranking the results by moving the focus of search through the cluster space (see Figure 8). The goal of the tool is to enhance a search engine by providing the users multiple rankings of the results for each query. It is achieved by generating topics for the given query and its result set, and visualizing these topics on the "Ranking Space" panel. When the focus is set near a given topic, results that are on or closer to that topic are ranked higher.

The figure shows a ranked list of relevant publications on the topic of "smells", in the period from 1900 to 1925. The list is modified by changing the context on the right side - the focus is changed by placing the cursor over a cluster, and publications associated with this cluster are displayed.

4. CONCLUSION

In this paper we demonstrated an approach towards observing olfactory related information in scientific publications, as recorded in the MAG.

In addition, we present an Odor Observatory tool that enables several usage scenarios for exploring historical and present olfactory research.

The future work will include the exploration of other textual datasets applicable for olfactory research, with an accent on olfactory heritage information.

In line with the Odeuropa project, the relevant information extracted from textual sources will be, following semantic web standards, aligned with the 'European Olfactory Knowledge Graph' (EOKG).

5. ACKNOWLEDGMENTS

This research is supported by the Slovenian research agency and by the European Union's Horizon 2020 program project Odeuropa under grant agreement number 101004469.

REFERENCES

[1] Wolfe, J. M., Kluender, K. R., Levi, D. M., Bartoshuk, L. M., Herz, R. S., Klatzky, R., Lederman, S. J., & Merfeld, D. M. (2012). Sensation & perception (3rd ed.). Sinauer Associates.

[2] Odeuropa project, https://odeuropa.eu (accessed in August, 2021).

[3] Wang K. et al. A Review of Microsoft Academic Services for Science of Science Studies, Frontiers in Big Data, 2019, doi: 10.3389/FDATA.2019.00045.

[4] JSI Odor Observatory, public service, https://odeuropa.ijs.si/dashboards/Main/Index?visualization=visualizations-MAG--top-topics# (accessed in August, 2021).

[5] Lötsch, J., Kringel, D., Hummel, T. Machine Learning in Human Olfactory Research, Chemical Senses, Volume 44, Issue 1, January 2019, Pages 11–22, https://doi.org/10.1093/chemse/bjy067.

[6] Iatropoulos, G., Herman, P., Lansner, A., Karlgren, J., Larsson, M., Olofsson, JK. The language of smell: Connecting linguistic and psychophysical properties of odor descriptors. Cognition. 2018 Sep;178:37-49. doi: 10.1016/j.cognition.2018.05.007. Epub 2018 May 12. PMID: 2976379.

[7] Tonelli, S. A Smell is Worth a Thousand Words: Olfactory Information Extraction and Semantic Processing in a Multilingual Perspective. doi: https://doi.org/10.4230/OASIcs.LDK.2021.2 https://drops.dagstuhl.de/opus/volltexte/2021/14538/pdf/OASIcs-LDK-2021-2.pdf (accessed in August, 2021).

Understanding Text Using Agent Based Models

Adrian Mladenic Grobelnik
Jozef Stefan Institute
Ljubljana Slovenia
adrian.m.grobelnik@ijs.si

Marko Grobelnik
Jozef Stefan Institute
Ljubljana Slovenia
marko.grobelnik@ijs.si

Dunja Mladenic
Jozef Stefan Institute
Ljubljana Slovenia
dunja.mladenic@ijs.si

ABSTRACT

The paper proposes a novel approach to text understanding and text generation focusing on short stories. The proposed approach attempts to understand and generate stories by creating an explainable, agent-based world model of the story. The world model is defined through agents, their goals, actions, attributes and relationships between them. We demonstrate our approach on the story of 'Little Red Riding Hood', simulating it as a sequence of 48 actions, involving 7 main agents and 14 goals.

KEYWORDS

Text understanding, agent-based approach, world model, agent-based model

1 Introduction

With recent advancements in deep learning and overall increases in computing power, artificial intelligence systems are now able to make commonsense inferences from simple events, as proposed in research such as COMET [1] and MultiCOMET [2]. While the aforementioned commonsense inferences can be made with a high degree of precision, they lack an explainable and comprehensive structure capable of storing and predicting future events with such inferences. Agent-based models (ABMs), while capable of simulating complex interactions between agents, rarely focus on understanding stories in greater depth. Moreover, they cannot perform commonsense reasoning on agent's goals, actions or attributes. In our research, we draw from existing work on ABMs to create a system capable of understanding short text-based stories, with the potential to incorporate commonsense inferences in the future.

Related work such as 'Automated Storytelling via Causal, Commonsense Plot Ordering' [3] and 'Modeling Protagonist Emotions for Emotion-Aware Storytelling' [4] makes use of COMET to tackle automated story plot generation. As the stories are generated using COMET's commonsense causal inferences, they lack explainability. In our work, we focus on generating explainable stories.

Other related work [5] focuses on story understanding using manually supplied commonsense rules, concept patterns and story text. Our system aims to understand and simulate a story, given the story text, goals and initial attributes of its agents.

Information Society 2021, 4-8 October 2021, Ljubljana, Slovenia

The main contributions of this paper are (1) a novel approach to explainable story understanding, (2) a system generating stories given a set of agents with attributes and goals, and (3) implementation of the proposed approach, with publicly available source code [7] allowing users to create and analyze their own stories.

The rest of this paper is organized as follows: Section 2 provides a problem description. Section 3 describes the approach used to tackle the problem. Section 4 demonstrates the functioning of our approach. The paper concludes with discussion and directions for future work in Section 5.

2 Problem Description

The problem we are solving is, given the text of a short story, convert it into a machine understandable and actionable description representing the dynamics of the story being told. Such an actionable description should encode the implicit knowledge assumed by the text in the form of an agent-based world model.

The world model should include enough representational power to fully represent the story. This includes agents, their environment and the relationships between them. The world model should be actionable enough to simulate the dynamics of an input story with all the key elements, and relevant details mentioned in the input text.

As the world model can represent a story given its text, it should also be able to represent and simulate other stories within the world model's constraints.

Some of the key operations the resulting system should support:

1. representation of the story
2. simulation of the story's dynamics
3. question answering about explicit and implicit elements written or assumed within the story
4. creating alternative stories, given their context

3 Approach Description

The general aim of our approach is to provide deep text understanding of the input story. Not all the steps are automatable at this stage. In particular, the biggest challenge is to automatically translate the story text into the knowledge based representation aligned with the world model. We are looking forward to eventually automate all of the steps in the approach.

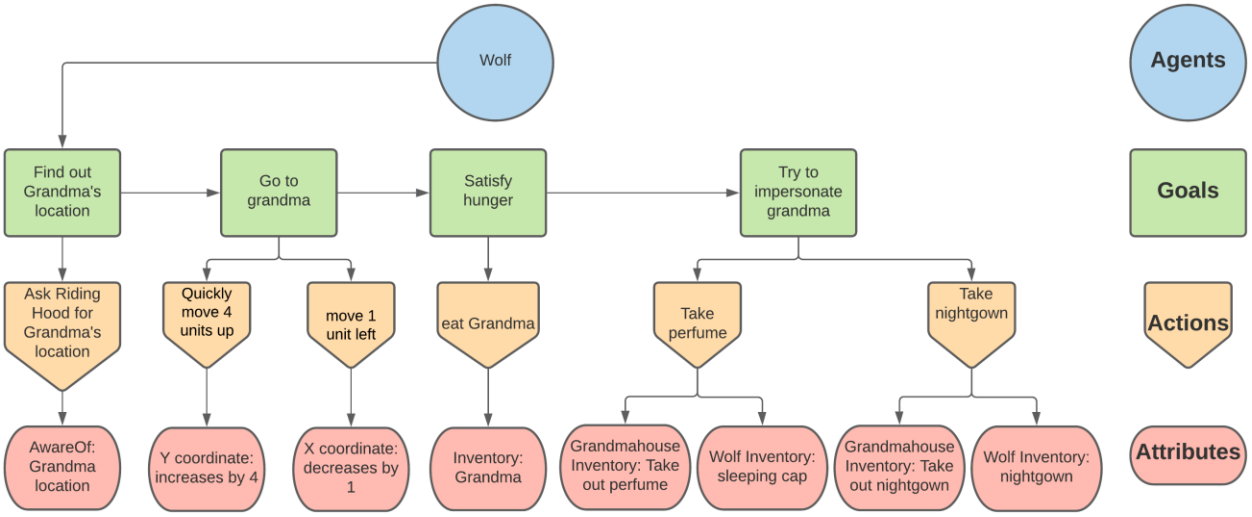

Figure 1: A partial representation of the Wolf agent's goals, actions and attributes.

As a running example of the input story, we selected the popular children's story 'Little Red Riding Hood' [6]. In the first stage, we restructured the original story into 73 simplified sentences where we identified 23 key events involving 7 main agents:

1. Mother
2. Riding Hood
3. Flower Field
4. Butterfly
5. Wolf
6. Grandma
7. Woodsman

Each agent is represented by its goals, actions and attributes (see Figure 1 for an example involving the Wolf). All goals cause actions and all actions change at least one agent's attributes.

As depicted on Figure 2, an agent's goal is defined by a goal state (a set of agents with specific attribute values) and 'pre-goals' (goals that must be completed and act as preconditions for an agent to start working towards the goal).

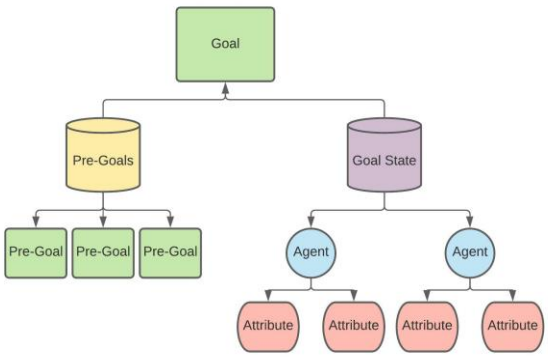

Figure 2: An example representation of a goal

To define actions, we use an action schema proposed as part of 'UCPOP: A Sound, Complete, Partial Order Planner for ADL' [8] where each action consists of a set of parameters, preconditions and effects. We show two example action representations in Figure 3 and Figure 4. The duration of each action corresponds to the passing of one time unit.

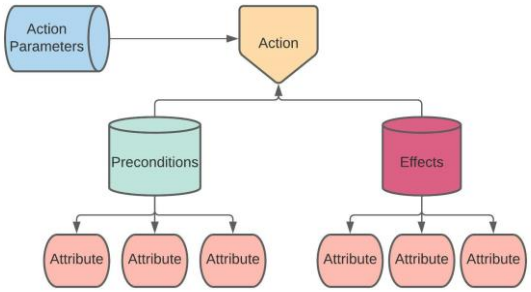

Figure 3: An example representation of an action

```
     action: Eat (?monster, ?victim, ?location)
precondition: knows(?monster, ?victim),
              alive(?monster), alive(?victim),
              ¬eaten( ?victim),¬full(?monster),
              at(?monster, ?location),
              at(?victim, ?location),
              ?monster ≠ ?victim
     effect: eaten(?victim)
              in(?victim, ?monster), full(?monster),
              ¬at(?victim, ?location)
```

Figure 4: An example pseudocode representation of a concrete action, taken from [9]

An attribute is simply defined as any information relating to the agent. For instance, the agent's location, inventory of items and awareness of other agents.

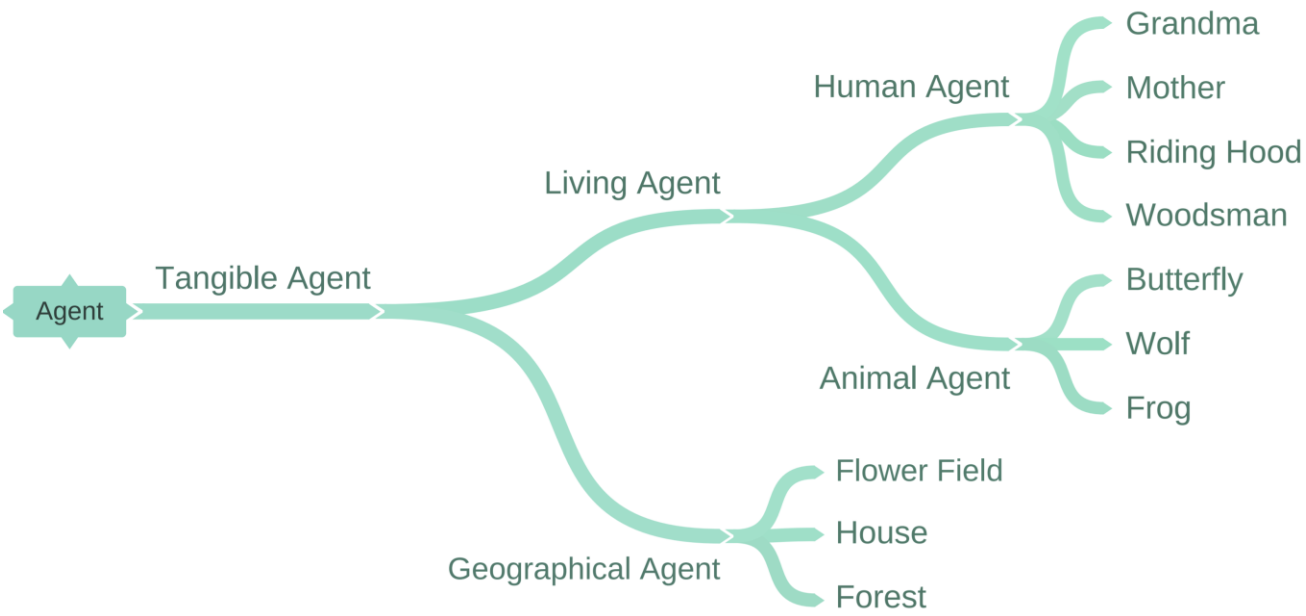

Figure 5: Hierarchy of agents for the Little Red Riding Hood story

The agents are defined through a hierarchy, ensuring consistency across agent goals, actions, attributes and providing a clear overview of the agent types as observed in Figure 5.

Throughout the story simulation of 'Little Red Riding Hood' 3 key agents jointly had 14 goals, causing them to perform a total of 48 actions composed of 12 unique action types.

We propose a simple textual description of each performed action, stating why the agent executed the action and which other agents were involved. See Figure 8 for an example.

At the highest conceptual level, we randomly select an agent and simulate all of its possible next actions. We then select the action that brings the agent closest to all it's currently active goals, and execute this action. We repeat this until there are no more agents with active goals in our world model, as depicted in Figure 6.

```
Until All Active Goals Not Complete
  For Random Agent
    For All Possible Actions
      Simulate Action
      If Action brings Agent closest to its Goals
        Set this as new Best Action
    Execute the Best Action
```

Figure 6: High level pseudocode of the simulation within the world model

4 Approach Demonstration

```
******************|grandmahouse,grandma|***********forest13|*******forest14,wolf
*************forest1|********************|*************forest5|*************forest12
****forest4,woodsman|********************|*************forest8|************forest10
*************forest3|********************|*************forest7|************forest11
*************forest2|********************|***forest6,butterfly|*************forest9
******flower field 1|******flower field 2|******flower field 3|********************
********************|********************|********************|********************
********************|********************|********************|********************
********************|******rhhouse,mother|********************|*********riding hood
```

Figure 7: Initial state of the agents' locations within the world model; each X, Y slot includes a list of agents at that location

We first initialize the world model to an initial setting similar to that of 'Little Red Riding Hood', illustrated in Figure 7. For instance, agents 'forest4' and 'woodsman' are in the same location, 1 unit above agent 'forest3'. The model is initialized with the agents, their initial attributes with values and their goals in the story. Once initialized, we can run the model and see the agents interact with each other within their environment. For an example, see Figure 9.

One could divide the story into the following 5 main segments:

1. Riding Hood discusses visiting Grandma with Mother **(6 actions)**
2. Riding Hood meets Wolf and goes to Grandma **(23 actions)**
3. Wolf eats Grandma and tries to impersonate her; Riding Hood arrives at GrandmaHouse and cries for help **(6 actions)**
4. Woodsman saves Grandma and takes Wolf away, Riding Hood gifts Grandma **(13 actions)**

As an example, in the third story segment the actions occur in the following order:

1. Wolf eats Grandma to satisfy hunger.
2. Wolf took perfume from GrandmaHouse's inventory to try impersonating Grandma.
3. Wolf took nightgown from GrandmaHouse's inventory to try impersonating Grandma.
4. Wolf took sleeping cap from GrandmaHouse's inventory to try impersonating Grandma.
5. Riding Hood moved 1 unit up to visit Grandma.
6. Riding Hood cried for help to get help.

The system is able to automatically generate the textual description of the story simulation over time, as depicted in Figure 8.

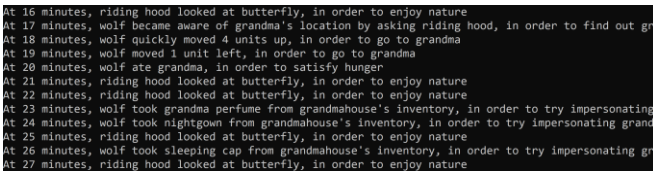

Figure 8: A part of an example story, generated by the system

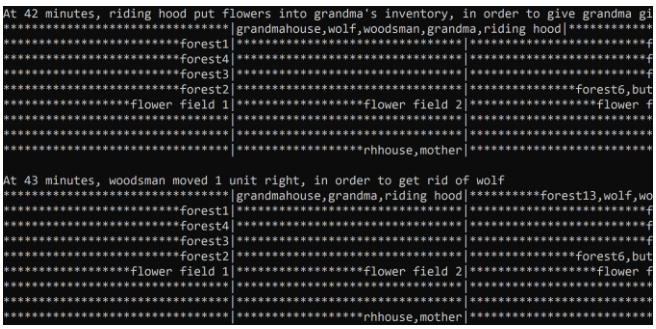

Figure 9: Screenshot of two subsequent agent location configurations on the map: (1) after Riding Hood gives Grandma flowers and (2) after Woodsman carries away Wolf

One of the more conceptually complex parts of the story was Riding Hood asking Mother for permission to visit Grandma. This required the creation of a new attribute for human agents to describe their opinions of other agents' goals.

The most complex action implemented was "cry for help". This involved the creation of a new goal "respond to cry for help" for all human agents within a certain radius of the agent crying for help, provided they were conscious and able to respond.

The story ends when Riding Hood gives Grandma the flowers she picked and the basket Mother gave her, and Woodsman carries the Wolf "deep into the forest where he wouldn't bother people any longer" [6].

The system was implemented in about 3,000 lines of C++ code, available on GitHub [7].

5 Discussion

In our research we expanded on and adapted existing work on agent-based models, providing an alternate approach to text understanding and generation involving short stories. As a proof of concept, we applied our approach on the children's story of 'Little Red Riding Hood', describing it through a series of 48 highly explainable actions involving 7 main agents.

Adapting the system to another story using our source code is relatively easy, provided the action and attribute types of the agents in the story are similar to those in the 'Little Red Riding Hood'. If the story requires the implementation of new actions or attributes, this can be done by extending the class structure in C++ using already implemented actions and attributes as examples.

In our future work we intend to integrate commonsense inferences, such as those from MultiCOMET into our model to further the system's degree of textual understanding. Our system could also benefit from the addition of dynamic and simultaneous goals that change based on the agent's environment. Another possible future line of work is to use our approach in other domains to describe more complex phenomena, such as real-world events or geopolitics. Lastly, a user evaluation of our system's performance on a variety of stories and scenarios could provide further insight into the efficacy of our approach.

ACKNOWLEDGMENTS

The research described in this paper was supported by the Slovenian research agency under the project J2-1736 Causalify and co-financed by the Republic of Slovenia and the European Union under the European Regional Development Fund. The operation is carried out under the Operational Programme for the Implementation of the EU Cohesion Policy 2014–2020.

REFERENCES

[1] Bosselut, A.; Rashkin, H.; Sap, M.; Malaviya, C.; Celikyilmaz, A.; and Choi, Y. 2019. COMET: Commonsense transformers for automatic knowledge graph construction. In ACL, 4762–4779.

[2] Adrian Mladenic Grobelnik, Marko Grobelnik, Dunja Mladenic. 2020. MultiCOMET – Multilingual Commonsense Description. In *Proceedings of the 23rd international multiconference information society, pages 37-40*

[3] Prithviraj Ammanabrolu, Wesley Cheung, William Broniec, and Mark O Riedl. 2021. Automated storytelling via causal, commonsense plot ordering. In Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI).

[4] Faeze Brahman and Snigdha Chaturvedi. 2020. Modeling protagonist emotions for emotion-aware storytelling. In Proceedings of EMNLP, pages 5277– 5294.

[5] Patrick Henry Winston. The genesis story understanding and story telling system: A 21st century step toward artificial intelligence. 2014. Technical report, Center for Brains, Minds and Machines (CBMM).

[6] Little Red Riding Hood by Leanne Guenther. https://www.dltk-teach.com/RHYMES/littlered/story.htm. Accessed 16.09.2021.

[7] Understanding Text Using Agent Based Models GitHub. https://github.com/AMGrobelnik/Understanding-Text-Using-Agent-Based-Models . Accessed 16.09.2021.

[8] Penberthy, J., & Weld, D. 1992. UCPOP: a sound, complete, partial-order planner for ADL. In Proceedings of KR'92, pp. 103–114, Los Altos, CA. Kaufmann.

[9] An Introduction to AI Story Generation. https://thegradient.pub/an-introduction-to-ai-story-generation/. Accessed 16.09.2021.

News Stream Clustering using Multilingual Language Models

Erik Novak
erik.novak@ijs.si
Jožef Stefan Institute
Jožef Stefan International Postgraduate School
Jamova cesta 39
Ljubljana, Slovenia

ABSTRACT

In this paper, we propose a news stream clustering algorithm which directly outputs cross-lingual event clusters. It uses multilingual language models to generate cross-lingual article representations which enable a direct comparison of articles in different languages. The algorithm is evaluated using a cross-lingual news article data set and compared against a strong baseline algorithm. The experiment results show the algorithm has great promise, but requires additional modifications for improving its performance.

KEYWORDS

online news, event detection, news events, multilingual language model

1 INTRODUCTION

Online news is producing hundreds of thousands of articles per day reporting about any significant event that happened in the world. The articles cover various domains (such as politics, sports, and culture) and are written in different languages. In order to automatically identify these events, news stream clustering algorithms are used. These usually have the following steps: (1) they group articles written in the same language into monolingual clusters, and (2) form cross-lingual clusters by linking monolingual clusters that report on the same event. Both steps usually employ monolingual text features such as TF-IDF vectors; these do not allow cross-lingual comparison without using advanced statistical or machine learning methods.

In this paper, we propose a news stream clustering algorithm that directly generates cross-lingual event clusters. The algorithm uses multilingual language models for generating cross-lingual content embeddings and extracting named entities found in the articles. These are used to measure if an article should be assigned to an event. The algorithm is evaluated using a cross-lingual data set consisting of articles in English, Spanish, and German, and is compared against a strong baseline. While the experiment results look promising, there is still room for improving the algorithms performance.

The paper is structured as follows: Section 2 contains an overview of the related work on cross-lingual news stream clustering and multilingual language models. Next, we present the proposed clustering algorithm in Section 3, and describe the experiment setting in Section 4. The experiment results are found in Section 5. Finally, we conclude the paper and provide ideas for future work in Section 6.

Information Society 2021, 4 - 8 October 2021, Ljubljana, Slovenia

2 RELATED WORK

News Stream Clustering. The objective of news stream clustering is to group news articles that report about the same event that happened in the world. Grouping can be a difficult task, especially if the articles are written in multiple languages. To this end, various approaches were developed for cross-lingual event clustering. A statistical approach called Generalization of Canonical Correlation Analysis is used to compare news articles in different languages [9]. Information extraction techniques, such as named entity recognition and part-of-speech tagging, are also used for event detection [6]. With the increasing popularity of neural networks, more advanced approaches are used to link event clusters. The work in [3] uses word embeddings to compare and link monolingual event clusters into cross-lingual ones. Transformer-based language models are used for event sentence coreference identification [4], a task that links parts of articles to multiple events. However, the algorithm is performed only on a monolingual data set.

To the best of our knowledge, our work is the first that uses multilingual language models for grouping articles directly into cross-lingual events.

Multilingual Language Models. Since the introduction of the transformers [11], language model development has gained traction in the research community. One of the most well known language models, BERT [2], has improved the performance of various NLP tasks. By training it using multilingual documents, the multilingual BERT [5] enabled solving tasks that require cross-lingual text representations. While these models improved the performance of various NLP tasks, they do not provide good document embeddings for tasks like clustering. This changed with the introduction of Sentence-BERT [8], which generates monolingual sentence embeddings appropriate for measuring sentence similarity. A year later, an approach for making monolingual document representations cross-lingual [7] opened a way for using sentence embeddings for cross-lingual clustering.

In this work, we employ the multilingual Sentence-BERT model to generate cross-lingual embeddings used to group articles into events.

3 THE CLUSTERING ALGORITHM

We propose a news stream clustering algorithm that directly outputs cross-lingual events. It uses cross-lingual embeddings, named entities, and temporal features to measure if an article should be assigned to an event cluster. If none of the events are appropriate, a new cluster is created and the article is assigned to it. Figure 1 shows the algorithm's workflow diagram.

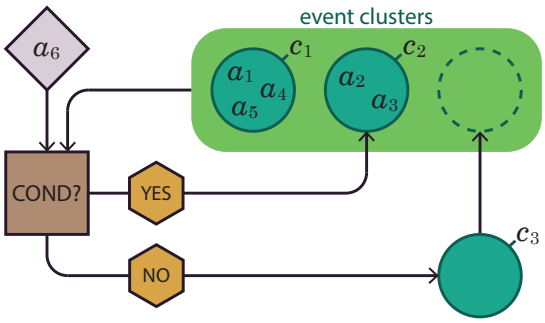

Figure 1: The algorithm's workflow diagram. The algorithm maintains a set of event clusters which are used when asessing if a new article (a_6) should be assigned to an existing event. If the conditions are met, the article is assigned to the most appropriate cluster (c_2). Otherwise, an empty event cluster is created (c_3), the article is assigned to it, and the newly created event is added to the cluster set.

In this section we describe how the algorithm represents the articles and events, and how it decides when to assign an article to the event cluster.

3.1 Article Representation

In this section we describe the different article representations used in the algorithm. Each article is assumed to have a title, body, and time attributes, which are used to (1) generate the content embedding and (2) extract its named entities.

Content Embedding. Each article is assigned an embedding that represents the article's content. Using multilingual Sentence-BERT[1], a language model designed for generating vectors used in cross-lingual clustering tasks, we get the content embedding by concatenating the article's title and body and inputing it into the language model. The output is a single 768 dimensional vector that captures the semantic meaning of the article.

Article Named Entities. For each article we extract the named entities that are mentioned in the article's body. To extract them, we developed a multilingual NER model using XLM-RoBERTa [1] and fine-tuned it using the CoNLL-2003 [10] data set.[2] Afterwards, we filter out the duplicates and store the remaining unique entities for later use.

3.2 Event Representations

An event is represented as an aggregate of its articles. This includes (1) the event centroid, (2) the named entities, and (3) the time statistics. In this section we describe how the aggregates are calculated and updated.

Event Centroid. The centroid represents the average content embedding of the articles assigned to the event. It is used to assess if an incoming article's content is similar enough to the event. Since the algorithm is intended to work on a news streams, we iteratively update the centroid with the newly assigned article's content embedding:

$$\vec{c_e}^{(0)} = \vec{0},$$
$$\vec{c_e}^{(k)} = \frac{(k-1)\cdot \vec{c_e}^{(k-1)} + \vec{c_{a_k}}}{k},$$

where $\vec{c_e}^{(k)}$ is the centroid calculated using the first k articles assigned to the event e, and $\vec{c_{a_k}}$ is the content embedding of the k-th article a_k.

Event Named Entities. Each event stores all of the unique named entities that are found in any of its articles. The named entities are used to identify if the incoming article mentions the event's entities. The event's named entities set is updated when a new article is assigned to the event:

$$r_e^{(0)} = \emptyset,$$
$$r_e^{(k)} = r_e^{(k-1)} \cup r_{a_k},$$

where $r_e^{(k)}$ is the set of named entities generated using the first k articles assigned to the event e, and r_{a_k} is the set of named entities of the k-th article a_k.

Time Statistics. The time statistics provide insights into the articles' temporal distribution. These are calculated using the articles' *time* attribute. In this experiment we measured the following statistics: the minimum, average, and maximum article timestamps. These are used to validate if an article was published at a time when it could still report about an existing event.

3.3 Assignment Condition

The most crucial part of the proposed algorithm is how to measure to which event should an article be assigned to, if any. We propose a condition that combines (1) the cosine similarity between the article's content embedding and the event's centroid, (2) the overlap between the article's and event's named entities, and (3) the time difference between the article's time and one of the event's time statistics.

Let $E = \{e_1, e_2, \ldots, e_j\}$ be the set of existing event clusters, where each event is represented with its centroid, named entities, and one of its time statistics $e_i = (\vec{c_{e_i}}, r_{e_i}, t_{e_i})$. Let the article be represented by its content embedding, named entities, and time attribute $a = (\vec{c_a}, r_a, t_a)$. We then check if the following conditions are met for each event:

$$\begin{aligned} \delta_c &= \frac{\langle \vec{c_{e_i}}, \vec{c_a} \rangle}{\|\vec{c_{e_i}}\|_2 \|\vec{c_a}\|_2} \geq \alpha, \\ \delta_r &= |r_{e_i} \cap r_a| \geq \beta, \\ \delta_t &= |t_{e_i} - t_a| \leq \tau, \end{aligned} \tag{1}$$

where α, β and τ are the thresholds corresponding to how similar the article's content must be to the event, the required amount of overlapping entities, and the time window in which an article has to be to be assigned to the event, respectively. Thus, $\delta_c, \delta_r, \delta_t$ correspond to the content similarity, entity overlap, and time window conditions, respectively.

If an event meets the conditions described in Equation 1, the article is assigned to it. If multiple events are appropriate, the article is assigned to the event that has the greatest δ_c value. If none are appropriate, a new empty event cluster is created, the article is assigned to it, and the event representations are updated.

To compare the impact of the conditions, we implement multiple versions of the algorithm that use a different combination

[1]The model is available at https://huggingface.co/sentence-transformers/paraphrase-xlm-r-multilingual-v1.

[2]The code of the model is available at https://github.com/ErikNovak/named-entity-recognition.

of δ_c, δ_r, and δ_t conditions. Table 1 shows all of the algorithm versions compared in the experiment.

Table 1: The list of algorithm versions. Each algorithm uses a different combination of conditions.

Algorithm	condition combination
CONTENT	δ_c
CONTENT + NE	δ_c *and* δ_r
CONTENT + TS	δ_c *and* δ_t
CONTENT + NE + TS	δ_c *and* δ_r *and* δ_t

4 EXPERIMENTS

We now present the experiment setting. We introduce the data set and how it is prepared for the experiment. Next, we present the evaluation metrics. Finally, the baseline algorithm is described.

4.1 Data Set

To compare the algorithm performances we use the news article data sets acquired via Event Registry and prepared by [3] for the purposes of news stream clustering. These data sets are in three different languages (English, German, and Spanish), and consist of articles containing the following attributes:

- *Title.* The title of the article.
- *Text.* The body of the article.
- *Lang.* The language of the article.
- *Date.* The datetime when the article was published.
- *Event ID.* The ID of the event the article is associated with. It is used to measure the performance of the algorithms.

For the experiment, we merge the three data sets together to create a single cross-lingual news article data set. We extract their content embeddings and named entities, and sort them in chronological order, i.e. from oldest to newest. Table 2 shows the data set statistics.

Table 2: Data set statistics. For each language data set we denote the number of documents in the data set (# docs), the average length of the documents (avg. length), the number of event clusters (# clusters) and the average number of documents in the clusters (avg. size).

Language	# docs	avg. length	# clusters	avg. size
English	8,726	537	238	37
German	2,101	450	122	17
Spanish	2,177	401	149	15
Together	13,004	500	427	30

4.2 Evaluation Metrics

For the evaluation we use the same metrics as [3]. Let tp be the number of correctly clustered-together article pairs, let fp be the number of incorrectly clustered-together article pairs, and let fn be the number of incorrectly not-clustered-together article pairs. Then we report precision as $P = \frac{\text{tp}}{\text{tp+fp}}$, recall as $R = \frac{\text{tp}}{\text{tp+fn}}$, and the balanced F-score as $F_1 = 2 \cdot \frac{P \cdot R}{P+R}$. While precision describes how homogenous are clusters the, recall tells us the amount of articles that should be together but are actually found in different clusters.

4.3 Baseline Algorithm

The baseline algorithm used in the experiment is presented in [3]. It performs cross-lingual news stream clustering by first generating monolingual event clusters using TF-IDF subvectors of words, word lemmas and named entities of the articles. Afterwards, it merges monolingual into cross-lingual clusters using cross-lingual word embeddings to represent the articles. The algorithm compares two approaches when performing cross-lingual clustering:

- *Global parameter.* Using a global parameter for measuring distances between all language articles for cross-lingual clustering decisions.
- *Pivot parameter.* Using a pivot parameter, where the distances between every other language are only compared to English, and cross-lingual clustering decisions are made only based on this distance.

Since the baseline algorithm was already evaluated using the cross-lingual data set we are using the the experiment, we only report their performances from the paper.

5 RESULTS

In this section we present the experiment results. For all experiments we fix the values $\beta = 1$ and $\tau = 3$ days, and evaluate the algorithms using different values of α. In addition, all experiments use the event's minimum time statistic when validating the time condition δ_t.

Baseline Comparison. Table 3 shows the experiment results of the best performing algorithm on the evaluation data set. We report the best performing CONTENT + NE + TS algorithm which uses the content similarity threshold $\alpha = 0.3$.

Table 3: The algorithm performances. The best reported algorithm uses all three asssignment conditions.

Algorithm	F_1	P	R
Baseline (global)	72.7	89.8	61.0
Baseline (pivot)	84.0	83.0	85.0
CONTENT + NE + TS	72.2	79.7	66.0

While the proposed algorithm does not perform better than any of the baselines with respect to the F_1 score, our algorithm still shows promising results. Its performance is comparable to the baseline using the global parameter and also outperforms the baseline (global) recall by 5%, showing it is better at grouping articles.

Condition Analysis. We have analyzed the impact the conditions have on the algorithm's performance. For each algorithm version we run the experiments using different values of $\alpha \in \{0.3, 0.4, 0.5, 0.6, 0.7\}$, and measure the balanced F-score, precision, and recall, as well as the number of clusters it generated. Table 4 shows the condition analysis results. By analysing the results we come to two conclusions:

Increasing α increases precision, decreases recall, and generates a larger number of clusters. When α is bigger, the content condition δ_c requires the articles to be more similar to the event. This condition is met when the article's content embedding is close to the event's centroid. Since this has to hold for all articles in the event, then the articles that have high similarity are clustered together, increasing the algorithm's precision.

Table 4: The condition analysis results. The bold values represent the best performances on the data set.

Algorithm	α	# clusters	F_1	P	R
CONTENT	0.3	46	29.6	19.7	59.8
	0.4	234	51.6	46.2	58.4
	0.5	849	57.7	67.7	50.3
	0.6	1762	45.3	73.1	32.8
	0.7	3185	26.0	81.9	15.5
CONTENT	0.3	279	43.7	33.3	63.8
+ NE	0.4	648	52.9	55.8	50.3
	0.5	1168	56.5	67.4	48.6
	0.6	1939	45.1	73.6	32.5
	0.7	3254	25.9	82.3	15.4
CONTENT	0.3	344	58.8	63.2	55.0
+ TS	0.4	806	64.1	76.5	55.2
	0.5	1346	58.8	83.4	45.4
	0.6	2068	47.1	81.7	33.1
	0.7	3356	25.2	**84.8**	14.7
CONTENT	0.3	925	**72.2**	79.7	**66.0**
+ NE	0.4	1221	**72.2**	80.5	65.5
+ TS	0.5	1554	54.0	81.9	40.2
	0.6	2174	46.7	80.7	32.9
	0.7	3403	25.0	**84.8**	14.7

However, if the α is to large then the condition is too strong, thus similar articles can be split into multiple clusters, consequently decreasing recall and increasing the number of clusters the algorithm generates.

Algorithms with more conditions can achieve better performance. The algorithm's performance is increasing with added conditions. While the worst performance is achieved when only the content condition δ_c is used (CONTENT algorithm), the best is reached when all three conditions are used (CONTENT + NE + TS algorithm). The most significant contribution is provided by the time condition δ_t which drastically improves the F_1 score.

6 CONCLUSION

We propose a news stream clustering algorithm that directly generates cross-lingual event clusters. It uses multilingual language models to generate cross-lingual article representations which are used to compare with and generate cross-lingual event clusters. The algorithm was evaluated on a news article data set and compared to a strong baseline. The experiment results look promising, but there is still room for improvement.

In the future, we intend to modify the assignment condition and learn the condition parameters instead of manually setting them. Modifying the language models to accept longer inputs could better capture the articles semantic meaning. In addition, events from different domains are reported with different rates. Learning these rates and including them in the algorithm could improve its performance.

ACKNOWLEDGMENTS

This work was supported by the Slovenian Research Agency and the Humane AI Net European Unions Horizon 2020 project under grant agreement No 952026.

REFERENCES

[1] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8440–8451.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, 4171–4186.

[3] Sebastião Miranda, Artūrs Znotiņš, Shay B Cohen, and Guntis Barzdins. 2018. Multilingual clustering of streaming news. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Brussels, Belgium.

[4] Faik Kerem Örs, Süveyda Yeniterzi, and Reyyan Yeniterzi. 2020. Event clustering within news articles. In *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*, 63–68.

[5] Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 4996–5001.

[6] Xiaoting Qu, Juan Yang, Bin Wu, and Haiming Xin. 2016. A news event detection algorithm based on key elements recognition. In *2016 IEEE First International Conference on Data Science in Cyberspace (DSC)*. (June 2016), 394–399.

[7] Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 4512–4525.

[8] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: sentence embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, 3982–3992.

[9] Jan Rupnik, Andrej Muhic, Gregor Leban, Primoz Skraba, Blaz Fortuna, and Marko Grobelnik. 2016. News across languages - Cross-Lingual document similarity and event tracking. en. *J. Artif. Intell. Res.*, 55, (January 2016), 283–316.

[10] Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, 142–147.

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. Curran Associates Inc., Red Hook, NY, USA, 6000–6010.

SloBERTa: Slovene monolingual large pretrained masked language model

Matej Ulčar and Marko Robnik-Šikonja
University of Ljubljana, Faculty of Computer and Information Science
Ljubljana, Slovenia
{matej.ulcar,marko.robnik}@fri.uni-lj.si

ABSTRACT

Large pretrained language models, based on the transformer architecture, show excellent results in solving many natural language processing tasks. The research is mostly focused on English language; however, many monolingual models for other languages have recently been trained. We trained first such monolingual model for Slovene, based on the RoBERTa model. We evaluated the newly trained SloBERTa model on several classification tasks. The results show an improvement over existing multilingual and monolingual models and present current state-of-the-art for Slovene.

KEYWORDS

natural language processing, BERT, RoBERTa, transformers, language model

1 INTRODUCTION

Solving natural language processing (NLP) tasks with neural networks requires presentation of text in a numerical vector format, called word embeddings. Embeddings assign each word its own vector in a vector space so that similar words have similar vectors, and certain relationships between word meanings are expressed in the vector space as distances and directions. Typical static word embedding models are word2vec [19], GloVe [24], and fastText [1]. ELMo [25] embeddings are an example of dynamic, contextual word embeddings. Unlike static word embeddings, where a word gets a fixed vector, contextual embeddings ascribe a different word vector for each occurrence of a word, based on its context.

State of-the-art text representations are currently based on the transformer architecture [35]. GPT-2 [27] and BERT [5] models are among the first and most influential transformer models. Due to their ability to be successfully adapted to a wide range of tasks, such models are, somewhat impetuously, called foundation models [2, 17]. While GPT-2 uses the transformer's decoder stack to model the next word based on previous words, BERT uses the encoder stack to encode word representations of a masked word, based on the surrounding context before and after the word. Previous embedding models (e.g., ELMo and fastText) were used to extract word representations which were then used to train a model on a specific task. In contrast to that, transformer models are typically fine-tuned for each individual downstream task, without extracting word vectors.

Information Society 2021, 4–8 October 2021, Ljubljana, Slovenia

Successful transformer models typically contain more than 100 million parameters. To train, they require considerable computational resources and large training corpora. Luckily, many of these models are publicly released. Their fine-tuning is much less computationally demanding and is accessible to users with modest computational resources. In this work, we present the training of a Slovene transformer-based masked language model, named SloBERTa, based on a variant of BERT architecture. SloBERTa is the first such publicly released model, trained exclusively on the Slovene language corpora.

2 RELATED WORK

Following the success of the BERT model [5], many transformer-based language models have been released, e.g., RoBERTa [14], GPT-3 [3], and T5 [28]. The complexity of these models has been constantly increasing. The size of newer generations of the models has made training computationally prohibitive for all research organizations and is only available to large corporations. Training also requires huge amounts of training data, which do not exist for most languages. Thus, most of these large models have been trained only for a few very well-resourced languages, chiefly English, or in a massively multilingual fashion.

The BERT model was pre-trained on two tasks simultaneously, a masked token prediction and next sentence prediction. For the masked token prediction, 15% of tokens in the training corpus were randomly masked before training. The training dataset was augmented by duplicating the training corpus a few times, with each copy having different randomly selected tokens masked. The next sentence prediction task attempts to predict if two given sentences appear in a natural order.

The RoBERTa [14] model uses the same architecture as BERT, but drops the next sentence prediction task, as it was shown that it does not contribute to the model performance. The masked token prediction task was changed so that the tokens are randomly masked on the fly, i.e. a different subset of tokens is masked in each training epoch.

Both BERT and RoBERTa were released in different sizes. Base models use 12 hidden transformer layers of size 768. Large models use 24 hidden transformer layers of size 1024. Smaller-sized BERT models exist using knowledge distillation from pre-trained larger models [11].

A few massively multilingual models were trained on 100 or more languages simultaneously. Notable released variants are multilingual BERT (mBERT) [5] and XLM-RoBERTa (XLM-R) [4]. While multilingual BERT models perform well for the trained languages, they lag behind the monolingual models [36, 33]. Examples of recently released monolingual BERT models for various languages are Finnish [36], Swedish [16], Estonian [30], Latvian [37], etc.

The Slovene language is supported by the aforementioned massively multilingual models and by the trilingual CroSloEngual BERT model [33], which has been trained on three languages,

Croatian, Slovene, and English. No monolingual transformer model for Slovene has been previously released.

3 SLOBERTA

The presented SloBERTa model is closely related to the French Camembert model [18], which uses the same architecture and training approach as the RoBERTa base model [14], but uses a different tokenization model. In this section, we describe the training datasets, the architecture, and the training procedure of SloBERTa.

3.1 Datasets

Training a successful transformer language model requires a large dataset. We combined five large Slovene corpora in our training dataset. Gigafida 2.0 [13] is a general language corpus, composed of fiction and non-fiction books, newspapers, school textbooks, texts from the internet, etc. The Janes corpus [9] is composed of several subcorpora. Each subcorpus contains texts from a certain social medium or a group of similar media, including Twitter, blog posts, forum conversations, comments under articles on news sites, etc. We used all Janes subcorpora, except Janes-tweet, since the contents of that subcorpus are encoded and need to be individually downloaded from Twitter, which is a lengthy process, as Twitter limits the access speed. KAS (Corpus of Academic Slovene) [8] consists of PhD, MSc, MA, Bsc, and BA theses written in Slovene between 2000 and 2018. SiParl [23] contains minutes of Slovene national assembly between 1990 and 2018. SlWaC [15] is a web corpus collected from the .si top-level web domain. All corpora used are listed in Table 1 along with their sizes.

Table 1: Corpora used in training of SloBERTa with their sizes in billion of tokens and words. Janes* corpus does not include Janes-tweet subcorpus.

Corpus	Genre	Tokens	Words
Gigafida 2.0	general language	1.33	1.11
Janes*	social media	0.10	0.08
KAS	academic	1.70	1.33
siParl 2.0	parliamentary	0.24	0.20
slWaC 2.1	web crawl	0.90	0.75
Total		4.27	3.47
Total after deduplication		4.20	3.41

3.2 Data preprocessing

We deduplicated the corpora, using the Onion tool [26]. We split the deduplicated corpora into three sets, training (99%), validation (0.5%), and test (0.5%). Independently of the three splits, we prepared a smaller dataset, one 15th of the size of the whole dataset, by randomly sampling the sentences. We used this smaller dataset to train a sentencepiece model[1], which is used to tokenize and encode the text into subword byte-pair-encodings (BPE). The sentencepiece model trained for SloBERTa has a vocabulary containing 32,000 subword tokens.

3.3 Architecture and training

SloBERTa has 12 transformer layers, which is equivalent in size to BERT-base and RoBERTa-base models. The size of each transformer layer is 768. We trained the model for 200,000 steps (about 98 epochs) on the Slovene corpora, described in Section 3.1. The model supports the maximum input sequence length of 512 subword tokens.

SloBERTa was trained as a masked language model, using fairseq toolkit [22]. 15% of the input tokens were randomly masked, and the task was to predict the masked tokens. We used the whole-word masking, meaning that if a word was split into more subtokens and one of them was masked, all the other subtokens pertaining to that word were masked as well. Tokens were masked dynamically, i.e. in each epoch, a different subset of tokens were randomly selected to be masked.

4 EVALUATION

We evaluated SloBERTa on five tasks: named-entity recognition (NER), part-of-speech tagging (POS), dependency parsing (DP), sentiment analysis (SA), and word analogy (WA). We used the labeled ssj500k corpus [12, 6] for fine-tuning SloBERTa on each of the NER, POS and DP tasks. For NER, we limited the scope to three types of named entities (person, location, and organization). We report the results as a macro-average F_1 score of these three classes. For POS-tagging, we used UPOS tags, the results are reported as a micro-average F_1 score. For DP, we report the results as a labeled attachement score (LAS). The SA classifier was fine-tuned on a dataset composed of Slovenian tweets [20, 21], labeled as either "positive", "negative", or "neutral". We report the results as a macro-average F_1 score.

Traditional WA task measures the distance between word vectors in a given analogy (e.g., man : king ≈ woman : queen). For contextual embeddings such as BERT, the task has to be modified to make sense. First, word embeddings from transformers are generally not used on their own, rather the model is fine-tuned. Four words from an analogy also do not provide enough context for use with transformers. In our modification, we input the four words of an analogy in a boilerplate sentence "If the word [word1] corresponds to the word [word2], then the word [word3] corresponds to the word [word4]." We then masked [word2] and attempted to predict it using masked token prediction. We used Slovene part of the multilingual culture-independent word analogy dataset [32]. We report the results as an average precision@5 (the proportion of the correct [word2] analogy words among the 5 most probable predictions).

We compared the performance of SloBERTa with three other transformer models supporting Slovene, CroSloEngual BERT (CSE-BERT) [33], multilingual BERT (mBERT) [5], and XLM-RoBERTa (XLM-R) [4]. Where sensible, we also included the results achieved with training a classifier model using Slovene ELMo [31] and fastText embeddings.

We fine-tuned the transformer models on each task by adding a classification head on top of the model. The exception is the DP task, where we used the modified dep2label-bert tool [29, 10]. For ELMo and fastText, we extracted embeddings from the training datasets and used them to train token-level and sentence-level classifiers for each task, except for the DP. The classifiers are composed of a few LSTM layer neural networks. For the DP task, we used the modified SuPar tool, based on the deep biaffine attention [7]. The details of the evaluation process are presented in [34].

The results are shown in Table 2. The results of ELMo and fastText, while comparable between each other, are not fully comparable with the results of transformer models as the classifier training approach is different.

[1] https://github.com/google/sentencepiece

Table 2: Results of Slovene transformer models.

Model	NER	POS	DP	SA	WA
fastText	0.478	0.527	/	0.435	/
ELMo	0.849	0.966	**0.914**	0.510	/
mBERT	0.885	0.984	0.681	0.576	0.061
XLM-R	0.912	0.988	0.793	0.604	0.146
CSE-BERT	0.928	0.990	0.854	0.610	0.195
SloBERTa	**0.933**	**0.991**	0.844	**0.623**	**0.405**

On the NER, POS, SA, and WA tasks, SloBERTa outperforms all other models/embeddings. For the POS-tagging, the differences between the models are small, except for fastText, which performs much worse. ELMo, surprisingly, outperforms all transformer models on the DP task. However, it performs worse on the other tasks. SloBERTa performs worse than CSE-BERT on the DP task, but beats other multilingual models.

The success of ELMo on the DP task can be partially explained by the different tools used for training the classifiers. Further work needs to be done to fully evaluate the difference and success of ELMo embeddings on this task.

The performance on the SA task is limited by the low inter-annotator agreement [20]. The reported average of F_1 scores for positive and negative class is 0.542 for inter-annotator agreement and 0.726 for self-agreement. Using the same measure (average of F_1 for positive and F_1 for negative class), SloBERTa scores 0.667, and mBERT scores 0.593.

On the WA task, most models perform poorly. This is expected because very little context was provided on the input, and the transformer models need a context to perform well. SloBERTa significantly outperforms other models, not only because it was trained only on Slovene data, but largely because its tokenizer is adapted to only Slovene language and does not need to cover other languages.

5 CONCLUSIONS

We present SloBERTa, the first monolingual transformer-based masked language model trained on Slovene texts. We show that SloBERTa large pretrained masked language model outperforms existing comparable multilingual models supporting Slovene on four tasks, NER, POS-tagging, sentiment analysis, and word analogy. The performance on the DP task is competitive, but lags behind some of the existing models.

In further work we intend to compare improvement of BERT-like monolingual models over multilingual models for other languages.

The pre-trained SloBERTa model is publicly available via CLARIN.SI[2] and Huggingface[3] repositories. We make the code, used for preprocessing the corpora and training the SloBERTa, publicly available[4].

ACKNOWLEDGMENTS

The work was partially supported by the Slovenian Research Agency (ARRS) core research programmes P6-0411 and project J6-2581, as well as the Ministry of Culture of Republic of Slovenia through project Development of Slovene in Digital Environment (RSDO). This paper is supported by European Union's Horizon 2020 research and innovation programme under grant agreement No 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media).

[2]http://hdl.handle.net/11356/1397
[3]https://huggingface.co/EMBEDDIA/sloberta
[4]https://github.com/clarinsi/Slovene-BERT-Tool

REFERENCES

[1] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5, 135–146.

[2] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the opportunities and risks of foundation models. ArXiv preprint 2108.07258. (2021).

[3] Tom Brown et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*. Volume 33, 1877–1901.

[4] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186. DOI: 10.18653/v1/N19-1423.

[6] Kaja Dobrovoljc, Tomaž Erjavec, and Simon Krek. 2017. The universal dependencies treebank for Slovenian. In *Proceeding of the 6th Workshop on Balto-Slavic Natural Language Processing (BSNLP 2017)*.

[7] Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *Proceedings of 5th International Conference on Learning Representations, ICLR*.

[8] Tomaž Erjavec, Darja Fišer, and Nikola Ljubešić. 2021. The KAS corpus of Slovenian academic writing. *Language Resources and Evaluation*, 55, 2, 551–583.

[9] Darja Fišer, Tomaž Erjavec, and Nikola Ljubešić. 2016. Janes v0. 4: korpus slovenskih spletnih uporabniških vsebin. *Slovenščina 2.0: empirical, applied and interdisciplinary research*, 4, 2, 67–99.

[10] Carlos Gómez-Rodríguez, Michalina Strzyz, and David Vilares. 2020. A unifying theory of transition-based and sequence labeling parsing. In *Proceedings of the 28th International Conference on Computational Linguistics*, 3776–3793. DOI: 10.18653/v1/2020.coling-main.336.

[11] Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. TinyBERT: Distilling BERT for natural language understanding. (2020). arXiv: 1909.10351 [cs.CL].

[12] Simon Krek, Kaja Dobrovoljc, Tomaž Erjavec, Sara Može, Nina Ledinek, Nanika Holz, Katja Zupan, Polona Gantar, Taja Kuzman, Jaka Čibej, Špela Arhar Holdt, Teja Kavčič, Iza Škrjanec, Dafne Marko, Lucija Jezeršek, and Anja Zajc. 2019. Training corpus ssj500k 2.2. Slovenian language resource repository CLARIN.SI. (2019).

[13] Simon Krek, Tomaž Erjavec, Andraž Repar, Jaka Čibej, Spela Arhar, Polona Gantar, Iztok Kosem, Marko Robnik, Nikola Ljubešić, Kaja Dobrovoljc, Cyprian Laskowski, Miha Grčar, Peter Holozan, Simon Šuster, Vojko Gorjanc, Marko Stabej, and Nataša Logar. 2019. Gigafida 2.0: Korpus pisne standardne slovenščine. viri.cjvt.si/gigafida. (2019).

[14] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. ArXiv preprint 1907.11692. (2019).

[15] Nikola Ljubešić and Tomaž Erjavec. 2011. hrWaC and slWaC: Compiling web corpora for Croatian and Slovene. In *International Conference on Text, Speech and Dialogue*. Springer, 395–402.

[16] Martin Malmsten, Love Börjeson, and Chris Haffenden. 2020. Playing with Words at the National Library of Sweden – Making a Swedish BERT. ArXiv preprint 2007.01658. (2020).

[17] Gary Marcus and Ernest Davis. 2021. Has AI found a new foundation? *The Gradient*. 11 September 2021.

[18] Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: A tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7203–7219.

[19] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint 1301.3781*.

[20] Igor Mozetič, Miha Grčar, and Jasmina Smailović. 2016. Multilingual Twitter sentiment classification: the role of human annotators. *PLOS ONE*, 11, 5.

[21] Igor Mozetič, Miha Grčar, and Jasmina Smailović. 2016. Twitter sentiment for 15 european languages. Slovenian language resource repository CLARIN.SI. (2016). http://hdl.handle.net/11356/1054.

[22] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. Fairseq: a fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

[23] Andrej Pančur and Tomaž Erjavec. 2020. The siParl corpus of Slovene parliamentary proceedings. In *Proceedings of the Second ParlaCLARIN Workshop*, 28–34.

[24] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing EMNLP*, 1532–1543.

[25] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2227–2237. DOI: 10.18653/v1/N18-1202.

[26] Jan Pomikálek. 2011. *Removing boilerplate and duplicate content from web corpora*. PhD thesis. Masaryk university, Brno, Czech Republic.

[27] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. OpenAI blog. 2019 Feb 24. (2019).

[28] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21, 1–67.

[29] Michalina Strzyz, David Vilares, and Carlos Gómez-Rodríguez. 2019. Viable dependency parsing as sequence labeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 717–723. DOI: 10.18653/v1/N19-1077.

[30] Hasan Tanvir, Claudia Kittask, and Kairit Sirts. 2020. EstBERT: A pretrained language-specific BERT for Estonian. arXiv preprint 2011.04784. (2020).

[31] Matej Ulčar and Marko Robnik-Šikonja. 2020. High quality ELMo embeddings for seven less-resourced languages. In *Proceedings of the 12th Language Resources and Evaluation Conference, LREC 2020*, 4733–4740.

[32] Matej Ulčar, Kristiina Vaik, Jessica Lindström, Milda Dailidėnaitė, and Marko Robnik-Šikonja. 2020. Multilingual culture-independent word analogy datasets. In *Proceedings of the 12th Language Resources and Evaluation Conference*, 4067–4073.

[33] Matej Ulčar and Marko Robnik-Šikonja. 2020. FinEst BERT and CroSloEngual BERT: less is more in multilingual models. In *Proceedings of Text, Speech, and Dialogue, TSD 2020*, 104–111.

[34] Matej Ulčar, Aleš Žagar, Carlos S. Armendariz, Andraž Repar, Senja Pollak, Matthew Purver, and Marko Robnik-Šikonja. 2021. Evaluation of contextual embeddings on less-resourced languages. ArXiv preprint 2107.10614. (2021).

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.

[36] Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. 2019. Multilingual is not enough: BERT for Finnish. arXiv preprint arXiv:1912.07076. (2019).

[37] Artūrs Znotiņš and Guntis Barzdiņš. 2020. LVBERT: Transformer-based model for Latvian language understanding. In *Human Language Technologies–The Baltic Perspective: Proceedings of the Ninth International Conference Baltic HLT 2020*. Volume 328, 111.

Understanding the Impact of Geographical Bias on News Sentiment: A Case Study on London and Rio Olympics

Swati
swati@ijs.si
Jožef Stefan Institute,
Jožef Stefan International Postgraduate School
Ljubljana, Slovenia

Dunja Mladenić
dunja.mladenic@ijs.si
Jožef Stefan Institute,
Jožef Stefan International Postgraduate School
Ljubljana, Slovenia

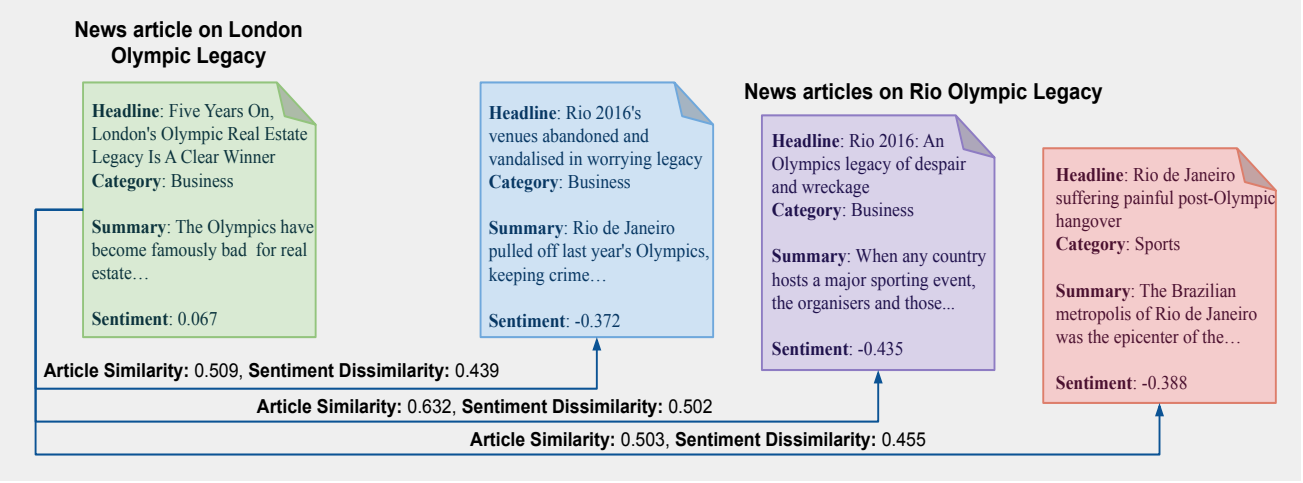

Figure 1: An example to illustrate the impact of geographical location on the sentiment of similar news articles.

ABSTRACT

There are various types of news bias, most of which play an important role in manipulating public perceptions of any event. Researchers frequently question the role of geographical location in attributing such biases. To that end, we intend to investigate the impact of geographical bias on news sentiments in related articles. As our case study, we use news articles collected from the Event Registry over two years about the Olympic legacy in London and Rio. Our experimental analysis reveals that geographical boundaries do have an impact on news sentiment.

KEYWORDS

Bias, News Bias, Geographical Bias, Olympics, Semantic Similarity, Sentiment Analysis, Dataset

1 INTRODUCTION

Claims of bias in news coverage raise questions about the role of geography in shaping public perceptions of similar events. Based on the geographical location, multiple factors, such as political affiliation, editorial independence, etc., can influence the way news articles are generated. Although it is well known that biased news can have more influence on people's thinking and decision-making processes [7, 9], it is nearly impossible to produce an article without any bias. Biased news articles have the potential to induce a variety of political and social implications, both direct and indirect. For instance, any political controversy presented from a specific perspective may alter the voting pattern [4, 1, 6].

There are different forms of news bias, and geographical bias is one of them. It exists if the sentiment polarity of similar articles published in different geographical location is contradictory or varies significantly. Sentiment analysis methods, which are commonly used to determine news bias [3, 14], can be used to examine the shift in sentiment polarity in similar news articles. Now, an intriguing question arises: Is geographical bias a factor affecting news sentiment? This study seeks to answer the above question by identifying and comparing sentiments of similar news articles. In doing so, we demonstrate how geographical location impacts the sentiments of similar articles. We also investigate this impact in relation to several news categories such as politics, business, sports, and so on.

The Olympic Games are a symbol of the greatest sports events in the world. Every edition leaves a number of legacies for the Olympic Movement, as well as unforgettable memories for each host city, whether positive or negative. In this regard, we select news articles about the Olympic legacy in London and Rio as a case study for our analysis.

We use Event Registry[1] [10] to collect English news articles, along with their sentiment and categories, published between January 2017 and December 2020. We use the popular Sentence-BERT (SBERT) [12] embedding to represent the articles and then compute the cosine similarity between them to identify similar article pairs.

Our data and code can be found in the GitHub repository at *https://github.com/Swati17293/geographical-bias*.

Information Society 2021, 4 October 2021, Ljubljana, Slovenia

[1] https://eventregistry.org

1.1 Contributions

The paper's contributions are as follows:

- We propose a task of analyzing the impact of geographical bias on the sentiment of news articles with data on the Olympic legacies of Rio and London as a case study.
- We present a dataset of English news articles customized to the above-mentioned task.
- We present experimental results to demonstrate the aforementioned impact of geographical bias.

2 RELATED WORK

The Majority of the sentiment analysis methods for news bias analysis depend on the sentiment words that are explicitly stated. SentiWordNet[2], which is a publicly available lexical resource used by the researchers for opinion mining to identify the sentiment inducing words that classify them as positive, negative, or neutral.

Melo et al. [5] collected and analyzed articles from Brazil's news media and social media to understand the country's response to the COVID-19 pandemic. They proposed using an enhanced topic model and sentiment analysis method to tackle this task. They identified and applied the main themes under consideration in order to comprehend how their sentiments changed over time. They discovered that certain elements in both media reflected negative attitudes toward political issues.

Quijote et al. [11] used SentiWordNet along with the Inverse Reinforcement Model to analyze the bias present in the news article and to determine whether the outlets are biased or not. The lexicons were first scored for the experiments using SentiWordNet and then fed to the Inverse Reinforcement model as input. To determine the news bias, the model measured the deviation and controversy scores of the articles. The findings lead to the inference that articles from major news outlets in the Philippines are not biased, excluding those from the Manila Times.

Bharathi and Geetha [3] classified the articles published by the UK, US, and India median as positive, negative, or neutral using the content sentiment algorithm [2]. The sentiment scores of the opinion words and their polarities were used as input to the algorithm.

Existing research investigates news bias using sentiment analysis methods, but, unlike our work, it does not provide a suitable automated method for analyzing the impact of geographical bias on news sentiment.

3 DATA DESCRIPTION

3.1 Raw Data Source

We use **Event Registry** [10] as our raw data source which monitors, gathers, and delivers news articles from all around the world. It also annotates articles with numerous metadata such as a unique identifier for article identification, categories to which it may belong, geographical location, sentiment, and so on. Its large-scale coverage can therefore be used effectively to assess the impact of geographical bias on news sentiment.

3.2 Dataset

To generate our dataset, we use a similar data collection process as described in [13]. Using the Event Registry API, we collect all English-language news articles about the Olympic legacy in London and Rio published between January 2017 and December 2020. We consider an article to be about the Olympic Legacy in London/Rio if the headline and/or summary of the article contains the keywords '*London*'/'*Rio*', '*Olympic*', and '*Legacy*'.

For each article, we then extract the summary, category, and sentiment. The article summaries vary in length from 290 to 6,553 words. Sentiment scores ranges from −1 to 1. We select seven major news categories, namely business, politics, technology, environment, health, sports, and arts-and-entertainment, and remove the rest of the categories. After excluding the duplicate articles we end up with 8,690 and 5,120 articles about the Olympic legacy in London and Rio respectively.

4 MATERIALS AND METHODS

4.1 Methodology

The primary task is to compute the average difference in sentiment scores between similar news articles about the Olympic legacies in Rio and London. The stated task can be subdivided and mathematically formulated as follows:

(1) Generate two distinct sets of news articles A_1 and A_2, one about the London Olympic legacy and the other about the Rio Olympic legacy. For each $a_i \in A_1$ find a list of $a'_j \in A_2$, where a_i is the i^{th} article in set $A_1 = \{(a_1, s_1), (a_2, s_2)...(a_n, s_n)\}$ and a'_j is the j^{th} article in set $A_2 = \{(a'_1, s'_1), (a'_2, s'_2)...(a'_m, s'_m)\}$ which is the closest match (c.f. Section 4.1.1) to a_i. Here, $n = |A_1|$ and $m = |A_2|$.
(2) For each list, calculate D_{ij} to represents the difference between the sentiment scores s_i and s'_j of the articles a_i and a'_j.
(3) Calculate the average difference D of sentiment scores.
(4) Calculate the percentage of similar article pairs with reversed polarity and those with unchanged polarity.

The secondary task is to assess the primary task with respect to news categories, i.e. to calculate the average difference D of sentiment scores for similar articles in each category.

In the following subsections, we discuss the tasks mentioned above in greater detail.

4.1.1 Article Similarity. We embed the articles in sets A_1 and A_2 to construct sets $F_1 = \{f_1, f_2...f_m\}$ and $F_2 = \{f'_1, f'_2...f'_n\}$. While alternative embedding approaches can be utilized, in this study we select the popular Sentence-BERT (SBERT) [12] embedding to extract 768-dimensional feature vectors to represent the individual articles in F_1 and F_2.

For each article a_i in A_1, we compute the similarity score[3] between a_i and every article a_j in A_2 using the cosine similarity metric $Sim^{cos}(a_i, a'_j)$ (Eq 1). We consider articles a_i and a'_j to be similar only if their similarity score is greater than 0.5.

$$Sim^{cos}(a_i, a'_j) = \frac{f_i \cdot f'_j}{||f_i||\ ||f'_j||} \tag{1}$$

where f_i and f'_j represents the embedded feature vectors of article a_i and a'_j.

The similarity score ranges from −1 to 1, where −1 indicates that the articles are completely unrelated and 1 indicates that they are identical, and in-between scores indicate partial similarity or dissimilarity.

4.1.2 Average Sentiment Dissimilarity. For every pair of similar articles a_i and a'_j, we calculate the difference D_{ij} between their sentiment scores s_i and s'_j. To calculate the average sentiment

[2] http://sentiwordnet.isti.cnr.it/

[3] https://en.wikipedia.org/wiki/Cosine_similarity

Table 1: Category-wise confusion matrix to show the percentage of similar article pairs with respect to their sentiment polarity.

	Sports		Business		Politics		Environment		Health		Technology		Arts & Entertainment	
	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
Pos	77	10	62	28	42	18	55	18	29	12	87	4	59	16
Neg	11	2	7	4	23	16	14	12	12	46	1	0	7	18

Table 2: Confusion matrix to show the percentage of similar article pairs with respect to their sentiment polarity.

	Positive	Negative
Positive	69	15
Negative	11	4

Table 3: Distribution of average sentiment difference across news categories for similar article pairs with identical category.

News category	Average Sentiment Difference
Sports	0.19
Business	0.20
Politics	0.18
Health	0.16
Environment	0.22
Technology	0.14
Arts and Entertainment	0.19

dissimilarity score D, we add all D_{ij} and divide it by the total number of similar article pairs.

5 RESULTS AND ANALYSIS

In our experiments, we compare 44,492,800 possible article pairs for similarity and discover 375,008 similar pairs. The comparison in terms of sentiment similarity reveals that if two articles from different geographical regions are similar, in our case Rio and London, the average difference in their sentiment scores is 0.171. In addition, as defined in Table 2, we calculate the percentage of similar article pairs based on their sentiment polarity. It's worth noting that the polarity of the article is completely reversed 27% of the time, indicating the impact of geographic region on sentiments.

It is because the success of mega-events such as the Olympics in a particular host city is heavily influenced by its residents' trust and support for the government [8]. It can be viewed positively as a national event with social and economic benefits, or negatively as a source of money waste. While the Olympics have left an economic and social legacy in London, a series of structural investment demands in Rio raise the question of whether or not the Olympics was worthwhile for the entire country.

5.1 Impact of news categories

The impact of news categories on the sentiments of similar articles with identical categories from different geographical regions is shown in Table 3. It demonstrates that certain news categories have a greater impact than others. Figure 2 depicts this distinction more clearly.

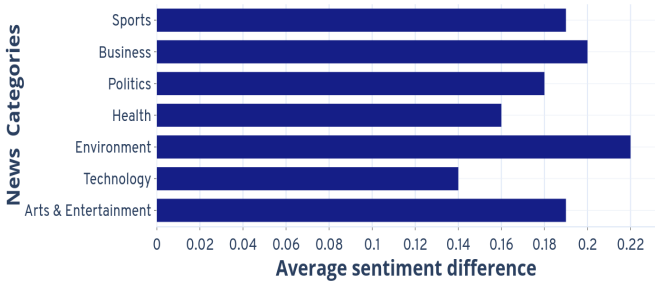

Figure 2: Distribution of average sentiment differences across categories for similar articles in the same category.

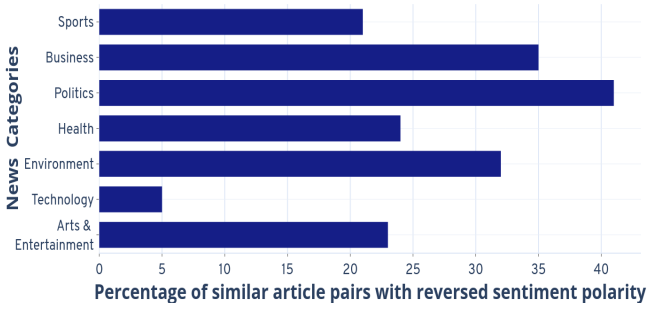

Figure 3: An illustration of the effect of category on sentiment polarity.

The categorical distribution of the percentage of similar article pairs in terms of sentiment polarity is shown in Table 1. *'Politics'* has the highest percentage of articles with reversed polarity, while *'technology'* has the lowest. Categories such as *'business'* and *'entertainment'*, though not as clearly as *'politics'*, exhibit the same bias.

This disparity arises from the fact that, in contrast to other categories, politics is most influenced by geographical boundaries, whereas science and technology are typically location independent. Since politics has such a large influence on shaping beliefs and public perceptions, it is frequently twisted to fit a particular narrative of a story. It is inherently linked to geographical borders, and it can be extremely polarizing depending on the geographical region.

6 CONCLUSIONS AND FUTURE WORK

In this work, we use news articles about the Olympic Legacy in London and Rio as a case study to understand how geographical boundaries interplay with news sentiments.

We begin by presenting a dataset of news articles collected over two years using the Event Registry API. We compute the cosine similarity scores of all possible embedded article pairs, one

from each set of Olympic legacy articles (London and Rio). We use the popular Sentence-BERT for article embedding and then compute the sentiment difference between similar article pairs. From 44,492,800 possible article pairs we end up with 375,008 similar pairs.

In our analysis, we discovered that the sentiment reflected in similar articles from different geographical regions differed significantly. We also investigate this difference in relation to different news categories such as politics, business, sports, and so on. We find a significant difference in news sentiment across geographical boundaries when it comes to political news, while in the case of news in technology, the difference is much smaller. We find that articles in categories such as politics and business can be heavily influenced by geographical location, articles in categories such as science and technology are typically location independent.

In the future, we plan to identify the most frequently mentioned topics in the Olympic legacy corpus to see how they affect the news sentiment of articles about different geographical locations. Since our study is limited to English news articles, we intend to learn more about the role of cultures and languages in this bias analysis. We also intend to broaden our investigation to discover the adjectives used to describe the negative and positive legacies of Rio and London. Such an analysis would aid in understanding the expectations from cities such as Rio (the first in South America to host the Olympics) in comparison to London.

7 ACKNOWLEDGMENTS

This work was supported by the Slovenian Research Agency and the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 812997.

REFERENCES

[1] Dan Bernhardt, Stefan Krasa, and Mattias Polborn. 2008. Political polarization and the electoral effects of media bias. *Journal of Public Economics*, 92, 5-6, 1092–1104.

[2] Shri Bharathi and Angelina Geetha. 2017. Sentiment analysis for effective stock market prediction. *International Journal of Intelligent Engineering and Systems*, 10, 3, 146–153.

[3] SV Shri Bharathi and Angelina Geetha. 2019. Determination of news biasedness using content sentiment analysis algorithm. *Indonesian Journal of Electrical Engineering and Computer Science*, 16, 2, 882–889.

[4] Chun-Fang Chiang and Brian Knight. 2011. Media bias and influence: evidence from newspaper endorsements. *The Review of economic studies*, 78, 3, 795–820.

[5] Tiago de Melo and Carlos MS Figueiredo. 2021. Comparing news articles and tweets about covid-19 in brazil: sentiment analysis and topic modeling approach. *JMIR Public Health and Surveillance*, 7, 2, e24585.

[6] Claes H De Vreese. 2005. News framing: theory and typology. *Information Design Journal & Document Design*, 13, 1.

[7] John Duggan and Cesar Martinelli. 2011. A spatial theory of media slant and voter choice. *The Review of Economic Studies*, 78, 2, 640–666.

[8] Dogan Gursoy and KW Kendall. 2006. Hosting mega events: modeling locals' support. *Annals of tourism research*, 33, 3, 603–623.

[9] Daniel Kahneman and Amos Tversky. 2013. Choices, values, and frames. In *Handbook of the fundamentals of financial decision making: Part I*. World Scientific, 269–278.

[10] Gregor Leban, Blaz Fortuna, Janez Brank, and Marko Grobelnik. 2014. Event registry: learning about world events from news. In *Proceedings of the 23rd International Conference on World Wide Web*, 107–110.

[11] TA Quijote, AD Zamoras, and A Ceniza. 2019. Bias detection in philippine political news articles using sentiwordnet and inverse reinforcement model. In *IOP Conference Series: Materials Science and Engineering* number 1. Volume 482. IOP Publishing, 012036.

[12] Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, (November 2019). http://arxiv.org/abs/1908.10084.

[13] Swati, Tomaž Erjavec, and Dunja Mladenić. 2020. Eveout: reproducible event dataset for studying and analyzing the complex event-outlet relationship.

[14] Taylor Thomsen. 2018. Do media companies drive bias? using sentiment analysis to measure media bias in newspaper tweets.

An evaluation of BERT and Doc2Vec model on the IPTC Subject Codes prediction dataset

Marko Pranjić
marko.pranjic@styria.ai
Jožef Stefan International Postgraduate School
Jamova cesta 39
Ljubljana, Slovenia
Trikoder d.o.o.
Zagreb, Croatia

Marko Robnik-Šikonja
marko.robnik@fri.uni-lj.si
University of Ljubljana, Faculty of Computer and Information Science
Ljubljana, Slovenia

Senja Pollak
senja.pollak@ijs.si
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia

ABSTRACT

Large pretrained language models like BERT have shown excellent generalization properties and have advanced the state of the art on various NLP tasks. In this paper we evaluate Finnish BERT (FinBERT) model on the IPTC Subject Codes prediction task. We compare it to a simpler Doc2Vec model used as a baseline. Due to hierarchical nature of IPTC Subject Codes, we also evaluate the effect of encoding the hierarchy in the network layer topology. Contrary to our expectations, a simpler baseline Doc2Vec model clearly outperforms the more complex FinBERT model and our attempts to encode hierarchy in a prediction network do not yield systematic improvement.

KEYWORDS

news categorization, text representation, BERT, Doc2Vec, IPTC Subject Codes

1 INTRODUCTION

The field of Natural Language Processing (NLP) has greatly benefited from the advances in deep learning. New techniques and architectures are developed at a fast pace. The Transformer architecture [12] is the foundation for most new NLP models and it is especially successful with models for text representation, such as BERT model [1] which dominates the text classification. The gains in performance promised by the large BERT models comes at the price of significant data resources and computational capabilities required in the model pretraining phase. The practitioners take one of the models pretrained in the language of the data and finetune it for the specific classification problem. Multilingual BERT-like models have also shown remarkable potential for cross-lingual transfer ([7], [8], [6]). A majority of the research with BERT-like models is focused on English, while less-resourced languages tend to be neglected.

The IPTC Subject Codes originate in the journalistic setting. The news articles are tagged with the IPTC topics to enable search and classification of the news content, as well as to facilitate content storage and digital asset management of news content at media houses. It provides a consistent and language agnostic coding of topics across different news providers and across time. Solving the automatic classification of the news content to the standardized set of topics would enable faster news production and higher quality of the metadata for news content.

In this paper, we use recently published STT News[10] dataset in Finnish to evaluate the performance of the monolingual FinBERT model [13] on the IPTC Subject Codes prediction task, together with the Doc2Vec[3] model as a baseline. We attempt to encode the hierarchical nature of the prediction task in the prediction network topology by mimicking the structure of the labels. Finally, impact of using a different tokenizers with the same model is evaluated.

The paper is structured as follows. In Section 2, we describe the dataset and the labels relevant for the prediction task. Section 3 describes the methods used to model the prediction task and all variations of experiments. In Section 4, we provide results of our experiments and, finally, in Section 5 we conclude this paper and suggest ideas for further work.

2 DATASET

The STT corpus [10] contains 2.8 million news articles from the Finnish News Agency (STT) published between 1992 and 2018. The articles come with a rich metadata information including the news article topics encoded as IPTC Subject Codes[1]. The IPTC Subject Codes are a deprecated version of IPTC taxonomy of news topics focused on text. The IPTC Subject Codes standard describes around 1400 topics structured in three hierarchical levels. The first level consists of the most general topics. Topics on the second level are subtopics of the ones at the first level and, likewise, topics on the third level are subtopics of the ones on second level. All topics on the third level are leaf topics - there are no more subdivisions, but there are also some topics on the second level that are leaf topics and do not extend to the third level. A set of IPTC topics at STT is an extended version of IPTC Subject Codes as some codes used at STT are not part of the IPTC standard.

Not all articles in the STT corpus contain the IPTC Subject Codes, as can be seen in Figure 1, showing the ratio of articles containing this information through time. IPTC Subject Codes were introduced in STT in May 2011 and around 10-15% of articles do not contain this information.

If an article contains a specific sub-topic, it also contains its upper-level topics. For example, if an article contains the third level topic "poetry", it also contains the second level topic "literature" that generalizes the "poetry", as well as the first level topic "arts, culture and entertainment". In this way, article metadata contains full path through the topic hierarchy.

Information Society 2021, 4–8 October 2021, Ljubljana, Slovenia

[1] https://iptc.org/standards/subject-codes/

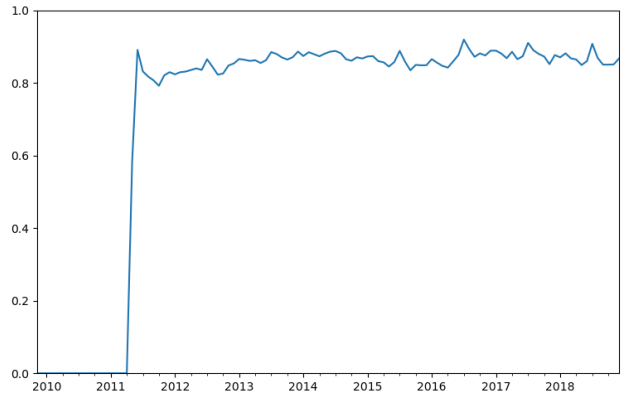

Figure 1: The Ratio of news articles in STT corpus containing IPTC Subject Codes.

Most articles are assigned only a small number of leaf-level topics (and its higher-level topics), but they can contain up to 7, 19 and 30 topics from the first, second and third level, respectively.

We split the dataset to train, validation and test set such that all articles published after 31-12-2017 belong to the test set and discard articles without IPTC Subject Codes from it. The rest of the articles were randomly split such that 5% of articles containing IPTC Subject Codes represent the validation set and all other articles belong to the train set.

After this step, there are around 30 thousand articles in the validation set, around 100 thousand in test set and 2.7 million in training set - of which some 560 thousand contain IPTC Subject Codes annotation.

The train set contains 17 different topics on the first level, 400 on the second level, and 972 on the third (the most specific) level. In our experiments, we evaluate models only on topics found in the training set.

3 METHODOLOGY

For our experiments, we used a network design consisting of two stacked neural networks (extractor and predictor). The extractor processes the text and produces the text representation in the format of a numeric vector. The predictor (the second part) is a multi-label prediction network that maps the extracted text representation vector to IPTC Subject Codes. For the extractor part, we evaluate the Doc2Vec and BERT model and for the predictor our models use one or three layer neural network.

3.1 Doc2Vec

Before the contextual token embeddings became popular, this model was regularly used to represent a text paragraph with a fixed vector. It was introduced in [3] with two variants of the algorithm - PV-DM (Paragraph Vector-Distributed Memory) and PV-CBOW (Paragraph Vector-Continuous Bag-of-Words). In the PV-DM variant of the algorithm, a training context is defined as a sliding window over the text. The model is a shallow neural network trained to predict the central word of this context window given the embeddings of the rest of the context words together with the embedding of the whole document. During training, the network learns both the word embeddings and the embedding for the document. The simpler PV-CBOW variant does not employ a context window, the neural network is trained to predict a randomly sampled word from the document. Our experiments use the PV-DM variant of the algorithm available in the Gensim[2] library with most of the hyperparameters set to their default values. We set the context window width to 5 and train the network for 10 epochs on the news content from the training data. The model produces a 256 dimensional output vector. Once the model is trained, we do not finetune it further during training of the prediction task.

Tokenization of the data was done using the SentencePiece[2] tokenizer. It was trained to produce a vocabulary of 40,000 tokens by using randomly selected 1 million sentences sampled from the articles in the training set. Additionally, we ran experiments using the same WordPiece[14] tokenizer that is used with the FinBERT model.

3.2 BERT

BERT is an deep neural-network architecture of bidirectional text encoders introduced in [1]. The base model consists of 12 Transformer [12] layers. It is trained using the masked language modeling (MLM) and next sentence prediction (NSP) objectives on a large text corpora. Maximum length of the input sequence for the model is 512 tokens and each token is represented with 768 dimensions. Model inference produces a context dependent representations of the input tokens. The whole input sequence can be represented with a single vector by using the context dependent representation of the *[CLS]* token. In [1], this representation is used as an aggregate sequence representation for classification tasks. Another way to represent the whole sequence, as used in [9], is to take the average representation of all output tokens (AVG). In this paper, we use FinBERT, a BERT model introduced in [13] that was pretrained on Finnish corpora.[3] We should note that this model contains the STT corpus as part of its training data.

Input to the model is restricted to 512 tokens[4] and longer news articles are trimmed such that only the first 512 tokens are used. In the dataset, there are less than 5% and 7% of documents in the training and test data that are longer than 512 tokens. We experiment with the CLS and the AVG representations and in both cases the article representation is a 768 dimensional vector. The FinBERT model is finetuned during training of the IPTC Subject Codes prediction task.

3.3 Prediction network

For the predictor part, we experiment with two different architectures. The first is a single layer of the neural network that maps the input vector to the predictions and can be seen in the Figure 2. The IPTC Subject Codes on all levels are concatenated together, thus producing a 1389 outputs in the final layer.

The second architecture utilizes the tree hierarchy of the IPTC Subject Codes. We assumed that a flat output (the previous approach) requires the network to predict each label independently, irrespective of the level of the target label. By introducing separate layers for each target level, we expect that the model will implicitly learn the hierarchy among labels. We designed this network in three layers and the architecture is shown in Figure 3. The first layer of the network predicts labels from the third IPTC hierarchical level (the most fine-grained topics), the second layer

[2]https://radimrehurek.com/gensim/
[3]We also test the FinEst BERT[11] but since the better performance was achieved with the FinBERT[13], we do not include FinEst BERT it in the results.
[4]The tokenizer used with the model is a predefined WordPiece tokenizer that came with the FinBERT model.

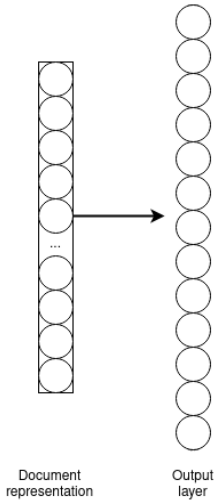

Figure 2: Predictor network architecture, flat variant. The image does not show a normalization layer before the output layer.

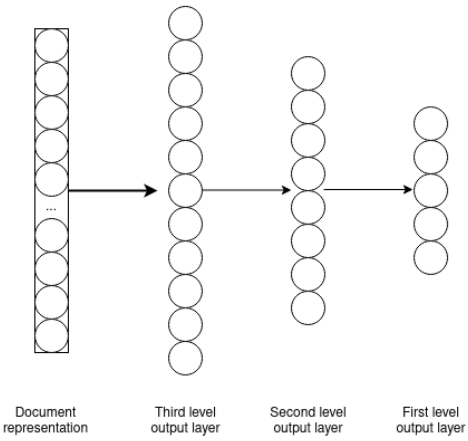

Figure 3: Predictor network architecture, tree variant. The image does not show a normalization layer before each output layer.

predicts topics from the second level and the third layer predicts only the toplevel IPTC Subject Codes.

3.4 Training

Each model was trained using the batch size of 128 articles and AdamW[4] optimizer with the learning rate of 1e-3. We compute the metrics on the validation set every 100 iterations. Once the loss on the validation data starts increasing, we stop the training and evaluate the best performing checkpoint on the test data. The loss function used in all experiments is the sum of binary cross-entropy losses calculated at each topic level. The news articles that do not have an annotation for certain topic level do not contribute to the loss of that level.

4 EXPERIMENTS AND RESULTS

All experiments were repeated three times and we report the median of those three runs in Table 1. The extraction network was evaluated with four configurations. The FinBERT model is using a WordPiece (WP) tokenizer and either the CLS token or the average (AVG) of all output tokens as a text representation. The Doc2Vec model is using either the WordPiece (WP) tokenizer or the SentencePiece (SP) tokenizer.

4.1 Evaluation metrics

We approach the article categorization problem through the information retrieval paradigm. Namely, we try to return the set of the most probable IPTC Subject Codes assigned to each article in the STT corpus. We use two performance metrics, the mean average precision (mAP) and recall at 10 (R@10). The mean average precision returns the expectation of the area under the precision-recall curve for a random query. The recall at 10 computes the ratio of correct topics found in the 10 tags with the highest predicted probability. To measure the generalization of our prediction models, we compute these metrics separately for each level of the IPTC Subject Codes.

4.2 Results and discussion

In all experiments, the Doc2Vec model performed significantly better than the FinBERT model, regardless of the specific extractor or predictor setup. This is surprising in the light of other successful applications of BERT models. Nevertheless, as there are less than 5% of articles in the training set and less than 7% of articles in the test set that have more than 512 tokens (the limitation of BERT but not Doc2Vec) we cannot assign the poor performance of BERT to this limitation.

Some other relevant findings are as follows. While for some tasks[9] the BERT average token representation performs better than the representation based on the CLS token, in our experiments the CLS and the AVG representations perform comparably. The three-layer network mimicking the shape of the tree-like IPTC Subject Codes hierarchy did not yield any systematic improvement over the single, flat layer of the neural network. Difference in tokenizers for Doc2Vec experiments shows small, but consistent improvement when using the SentencePiece tokenizer.

5 CONCLUSIONS AND FURTHER WORK

In this work, we have compared a monolingual FinBERT and Doc2Vec model on the IPTC Subject Codes prediction task in Finnish language. We evaluated several variations of experiments and achieved consistently better results with a Doc2Vec model. In contrast to the Doc2Vec, the BERT model has a limitation in the form of maximum number of input tokens. We believe the results cannot be explained by this as the data used does not contain a significant amount of documents exceeding this limit. We plan to explore this topic further in hope of understanding and addressing this problem. Recent work in BERT finetuning strategies[5] identifies a problem of vanishing gradients due to excessive learning rates and implementation details of the optimizer.

Our attempt at encoding the hierarchical nature of the prediction task did not yield systematic improvement and we believe it is worthwhile to explore other strategies and improve on this area, like encoding the hierarchy of the predictions in the loss function itself.

For Doc2Vec experiments, consistently better results were achieved using the SentencePiece[2] tokenizer over the WordPiece[14] tokenizer used in FinBERT model. Both of those tokenizers retain the whole information of the input as there are no destructive operations on the text. We plan further experiments

Table 1: Results for different experimental configurations.

Extractor	Predictor	mAP (lvl 1)	mAP (lvl 2)	mAP (lvl 3)	R@10 (lvl 1)	R@10 (lvl 2)	R@10 (lvl 3)
FinBERT (CLS)	Flat	0.5432	0.2047	0.1031	0.9058	0.3687	0.2242
FinBERT (CLS)	Tree	0.5434	0.1949	0.1043	0.9058	0.3602	0.2417
FinBERT (AVG)	Flat	0.5401	0.2026	0.1006	0.9045	0.3692	0.2391
FinBERT (AVG)	Tree	0.5410	0.2088	0.1089	0.9078	0.3724	0.2367
Doc2Vec (WP)	Flat	0.8091	0.5204	0.2990	0.9721	0.7008	0.4750
Doc2Vec (WP)	Tree	0.8127	0.5202	0.2972	0.9743	0.7099	0.4714
Doc2Vec (SP)	Flat	0.8298	0.5550	0.3149	0.9803	0.7277	**0.4951**
Doc2Vec (SP)	Tree	**0.8315**	**0.5643**	**0.3282**	**0.9832**	**0.7358**	0.4896

to confirm and quantify these findings and understand what enables such improvement of downstream prediction task at the tokenizer level.

ACKNOWLEDGMENTS

The work was partially supported by the Slovenian Research Agency (ARRS) core research programmes P6-0411 and P2-0103, as well as the research project J6-2581 (Computer-assisted multilingual news discourse analysis with contextual embeddings). This paper is supported by European Union's Horizon 2020 research and innovation programme under grant agreement No 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media).

REFERENCES

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. (June 2019), 4171–4186. DOI: 10.18653/v1/N19-1423.

[2] Taku Kudo and John Richardson. 2018. Sentencepiece: a simple and language independent subword tokenizer and detokenizer for neural text processing. In (January 2018), 66–71. DOI: 10.18653/v1/D18-2012.

[3] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning* (Proceedings of Machine Learning Research) number 2. Eric P. Xing and Tony Jebara, editors. Volume 32. PMLR, Bejing, China, (June 2014), 1188–1196. https://proceedings.mlr.press/v32/le14.html.

[4] Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*. https://openreview.net/forum?id=Bkg6RiCqY7.

[5] Marius Mosbach, Maksym Andriushchenko, and Dietrich Klakow. 2021. On the stability of fine-tuning {bert}: misconceptions, explanations, and strong baselines. In *International Conference on Learning Representations*. https://openreview.net/forum?id=nzpLWnVAyah.

[6] Andraž Pelicon, Marko Pranjić, Dragana Miljković, Blaž Škrlj, and Senja Pollak. 2020. Zero-shot learning for cross-lingual news sentiment classification. *Applied Sciences*, 10, 17. ISSN: 2076-3417. DOI: 10.3390/app10175993. https://www.mdpi.com/2076-3417/10/17/5993.

[7] Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, (July 2019), 4996–5001. DOI: 10.18653/v1/P19-1493. https://aclanthology.org/P19-1493.

[8] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21, 140, 1–67. http://jmlr.org/papers/v21/20-074.html.

[9] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, (November 2019), 3982–3992. DOI: 10.18653/v1/D19-1410. https://aclanthology.org/D19-1410.

[10] STT. 2019. Finnish news agency archive 1992-2018, source (http://urn.fi/urn:nbn:fi:lb-2019041501). (2019).

[11] Matej Ulčar and Marko Robnik-Šikonja. 2020. Finest bert and crosloengual bert. In *Text, Speech, and Dialogue*. Petr Sojka, Ivan Kopeček, Karel Pala, and Aleš Horák, editors. Springer International Publishing, Cham, 104–111.

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (NIPS'17). Curran Associates Inc., Red Hook, NY, USA, 6000–6010. ISBN: 9781510860964.

[13] Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. 2019. Multilingual is not enough: bert for finnish. (2019). arXiv: 1912.07076 [cs.CL].

[14] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: bridging the gap between human and machine translation. (2016). arXiv: 1609.08144 [cs.CL].

Classification of Cross-cultural News Events

Abdul Sittar*
abdul.sittar@ijs.si
Jožef Stefan Institute and Jožef Stefan International Postgraduate School
Jamova cesta 39
Ljubljana, Slovenia

Dunja Mladenić
dunja.mladenic@ijs.si
Jožef Stefan Institute and Jožef Stefan International Postgraduate School
Jamova cesta 39
Ljubljana, Slovenia

ABSTRACT

We present a methodology to support the analysis of culture from text such as news events and demonstrate its usefulness on categorising news events from different categories (society, business, health, recreation, science, shopping, sports, arts, computers, games and home) across different geographical locations (different places in 117 countries). We group countries based on the culture that they follow and then filter the news events based on their content category. The news events are automatically labelled with the help of Hofstede's cultural dimensions. We present combinations of events across different categories and check the performances of different classification methods. We also presents experimental comparison of different number of features in order to find a suitable set to represent the culture.

KEYWORDS

cultural barrier, news events, text classification

1 INTRODUCTION

Culture is defined as a collective programming of the mind which distinguishes the members of one group or category of people from another [9]. It has a huge impact on the lives of people and in result it influences events that involve cross-cultural stakeholders. News spreading is one of the most effective mechanisms for spreading information across the borders. The news to be spread wider cross multiple barriers such as linguistic, economic, geographical, political, time zone, and cultural barriers. Due to rapidly growing number of events with significant international impact, cross-cultural analytics gain increased importance for professionals and researchers in many disciplines, including digital humanities, media studies, and journalism. The most recent examples of such events include COVID-19 and Brexit [1]. There are few determinants that have significant influence on the process of information selection, analysis and propagation. These include cultural values and differences, economic conditions and association between countries. For instance, if two countries are culturally more similar, there are more chances that there will be a heavier news flow between them [10], [3]. In this paper, we focus on classification of news events across different cultures. We select some of the most read daily newspapers and collect information using Event Registry about the news they have published. Event Registry is a system which analyzes news articles, identifies groups of articles that describe the same event and represent them as a single event [7]. The description of the meta data of an event is shown in the Table 1. The main scientific contributions of this paper are the following:

(1) A novel perspective of aligning news events across different cultures through categorising countries and news events.
(2) A cross-cultural automatically annotated dataset in several different domains (Business, Science, Sports, Health etc.).
(3) Experimental comparison of several classification models adopting different set of features (character ngrams, GLOVE embeddings and word ngrams).

Table 1: The description of the meta data of an event.

Attributes	Description
title	title of the event
summary	summary of the event
source	event reported by a news source
categories	list of DMOZ categories
location	location of the event

2 RELATED WORK

In this section, we review the related literature about the influence of culture, its representation and classification in different fields.

Countries that share a common culture are expected to have heavier news flows between them when reporting on similar events [10]. There are many quantitative studies that found demographic, psychological, socio-cultural, source, system, and content-related aspects [2].

Cross-cultural research and understanding the cultural influences in different fields have competitive advantages. The goal of researching the impact of culture might be to draw conclusions in which way the cultural factors influence a specific corporate action. There are many type of cultures such as societal, organizational, and business culture etc [8].

The hidden nature of cultural behavior causes some difficulties in measurement and defining these. To cope with difficulties, researchers have developed measurements that measure culture on a general scale to compare differences among cultures and management styles. These results can be used to find similarities within a region and differences to other regions. There are many models that have tried to explain cultural differences between societies. Hofstede's national culture dimensions (HNCD) have been widely used and cited in different disciplines [6, 5]. Hofstede's dimensions are the result of a factor analysis at the level of country means of comprehensive survey instrument, aimed at identifying systematic differences in national cultural. Their purpose is to measure culture in countries, societies, sub-groups, and organizations; they are not meant to be regarded as psychological traits.

There is a plethora of research studies that were conducted to understand the cultural influences such as cross-culture privacy and

Information Society 2021, 4–8 October 2021, Ljubljana, Slovenia

attitude prediction, and cultural influences on today's business. [4] explores how culture affects the technological, organizational, and environmental determinants of machine learning adoption by conducting a comparative case study between Germany and US. Rather than looking at the influence of cultural differences within one domain, we intend to understand association between news events belonging to different domains (society, business, health, recreation, science, shopping, sports, arts, computers, games and home) and different cultures (117 countries from all the continents). We conduct this research to find an appropriate representation and classification of culture across different domains.

3 DATA DESCRIPTION

3.1 Dataset Statistics

We choose the top 10 daily read newspapers in the world in 2020 [1] and collect the events reported by these newspapers using Event Registry [7] over the time period of 2016-2020. Approximately 8000 events belongs to each newspaper with exception of "Zaman" that has only 900 events. Figure 1 shows the number of events reported by the selected newspapers on a yearly basis. This dataset can be found on the Zenodo repository (version 1.0.0) [2]

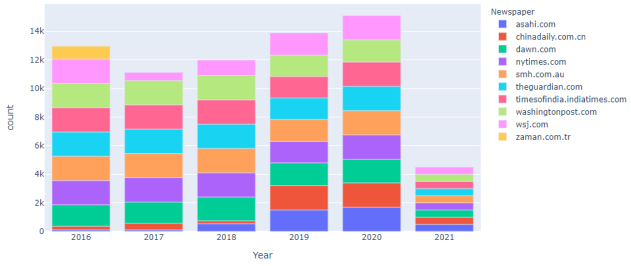

Figure 1: Each color in a bar represents the total number of events per year by a daily newspaper and a complete bar shows the total number of events per year by all the newspapers.

The attributes of an event with description are displayed in Table 1. Few attributes are self-explanatory such as title, summary, date, and source. DMOZ-categories are used to represent topics of the content. The DMOZ project is a hierarchical collection of web page links organized by subject matters [3]. Event Registry use top 3 levels of DMoz taxonomy which amount to about 50,000 categories [4].

4 MATERIAL AND METHODS

4.1 Problem Definition

There are two main parts of the problem that we are addressing. The first part is to label the examples by assigning a culture C to a news event E using its location L. The second part is a multi-class classification task where we predict the culture C of a news event E using its summary description S and its content category G as provided by the Event Registry. This task can be formulated as:

$$C = f(S, G)$$

C donates the culture of the news event, f is the learning function, S donates summary of a news event and G donates category of a news event (see Table 1).

4.2 Methodology

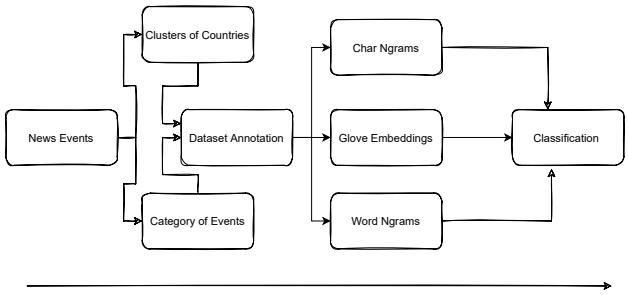

Figure 2: Classification of cross-cultural news events.

4.2.1 Data labeling. Each news event has information about the type of categories to which it belongs and the location where it happened (see Table 1). Each event has many categories and each category has a weight reflecting its relevance for the event. We only keep the most relevant categories and group the news events based on their categories. For each group of events, we estimate the cultural characteristic of each event through the country of the place where the event occurred. We cluster the countries based on their culture. We utilize the Hofstede's national culture dimensions (HNCD) to represent the culture of a country. We take average of cultural dimensions and call it average cultural score. Based on this score, we find optimal number of clusters using popular clustering algorithm k-means (see Figure 4). Finally, we label each news event with one of the six cultural clusters.

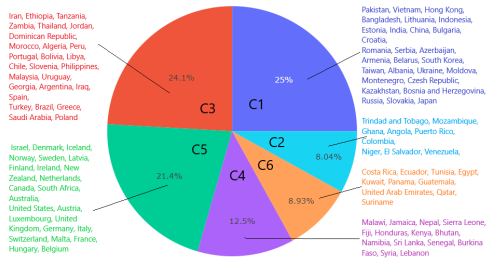

Figure 3: The pie chart depicts the percentage of the news events that occurred in six different clusters (each cluster consists of a list of countries with similar culture).

4.2.2 Data representation. Each news event in Event Registry has associated categories with it along with a weight (see Table 1), we take the top categories based on their weight. In case of multiple categories with equal weight, we sort them alphabetically and keep the first one. We represent each news event by a short summary S and a set of content categories G.

[1] https://www.trendrr.net/
[2] https://zenodo.org/record/5225053
[3] https://dmoz-odp.org/
[4] https://eventregistry.org/documentation?tab=terminology

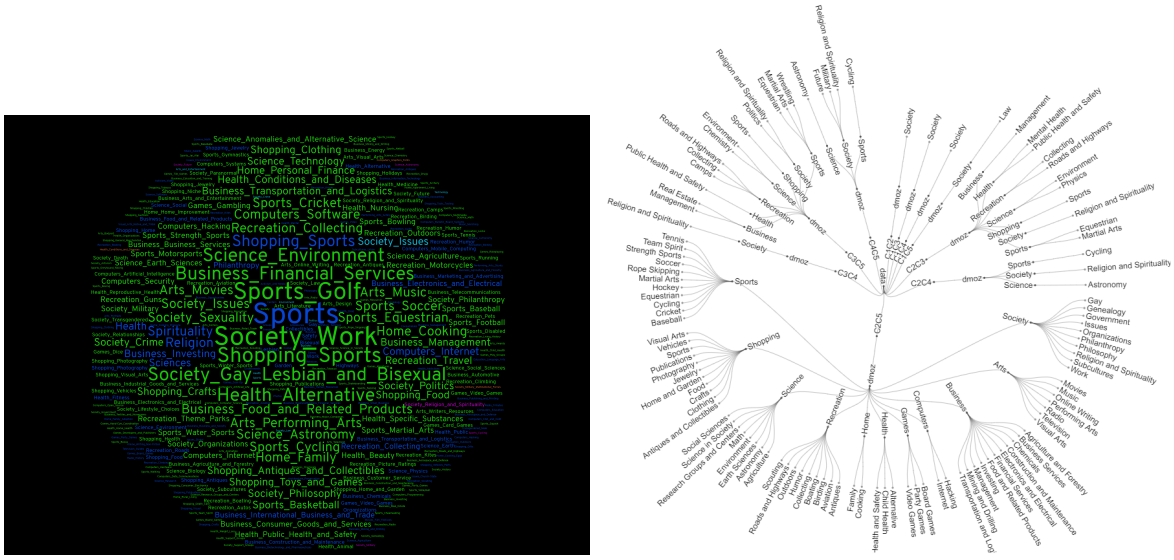

Figure 4: In word cloud, the color of each word shows cluster to whom it belongs (see Figure 3). Radial dendrograms illustrate the shared categories of news events between the pair of six clusters.

4.2.3 Data Modeling. For multi-class classification task, we use simple classification models (SVM, Decision Tree, KNN, Naive Bayes, Logistic Regression) as well as neural network. For simple classification models, we input character and word ngrams varying the number of ngrams and compare the results. We also use pre-trained Glove embeddings.

5 EXPERIMENTAL EVALUATION

5.1 Evaluation Metric

For multi-class classification task, we use following most commonly used evaluation measures: accuracy, precision, recall, and F1 score.

6 RESULTS AND ANALYSIS

6.1 Annotation Results

The results of annotation are six clusters where almost 50% news events belong to the two clusters (shown with red and blue colors) and remaining 50% belong to the other four clusters 3. Looking in each group, we find that clusters do not lies in a specific geographic area or a continent. Rather all the countries in a cluster belong to the different continents. Similarly, these clusters do not have all the countries that are economically rich or poor.

There are more categories in green and red colors in the word cloud (see Figure 4) which represent to the cluster with that colors. Radial dendrograms in Figure 4 present the shared categories between the clusters. In the figure, root of the tree is data and then there are ten pair of clusters that share the same categories. The objective of this whole process was to keep news events according to the category to whom they belongs. Moreover, we can only observe the cultural differences when we have same type of news events from different places.

6.2 Classification Results

Fro the experimental results we can see that the best performance is achieved by Logistic Regression, kNN and Decision Tree. The performance of SVM varies depending on the number of selected features: the highest F1-score is achieved with the top 10K or 20K word ngrams using 1 to 3 word ngrams (see Figure 5). Looking at the character ngrams, the highest F1-score is achieved when we select the top 15K characters for all the tested algorithms except Naive Bayes which declines in performance with the growing set of features. Based on these settings, we achieve the highest accuracy (0.85) using Logistic Regression. Using Glove embeddings, we experiment with and without using the category of event. The highest F1-score with and without the category is 0.80 and 0.79 respectively.

7 CONCLUSIONS AND FUTURE WORK

For researchers and professionals, it is very important to analyse the cross-cultural differences in different disciplines. As the international impact is increasing and international events are becoming popular, the need to develop some automatic methods is significantly increasing and leaving a blank space. We conducted experiments on news events related to different fields to have a broader look on data and machine learning methods. Further research would be helpful in examining the impact of specific socio-cultural factors on news events. In this research work, we estimate the culture of a specific place by its country, use basic features and simple classification models. To continue this work further, we would like to improve feature set such as by including part of speech tagging (POS) as well as other state of the art embeddings.

ACKNOWLEDGMENTS

The research described in this paper was supported by the Slovenian research agency under the project J2-1736 Causalify and by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 812997.

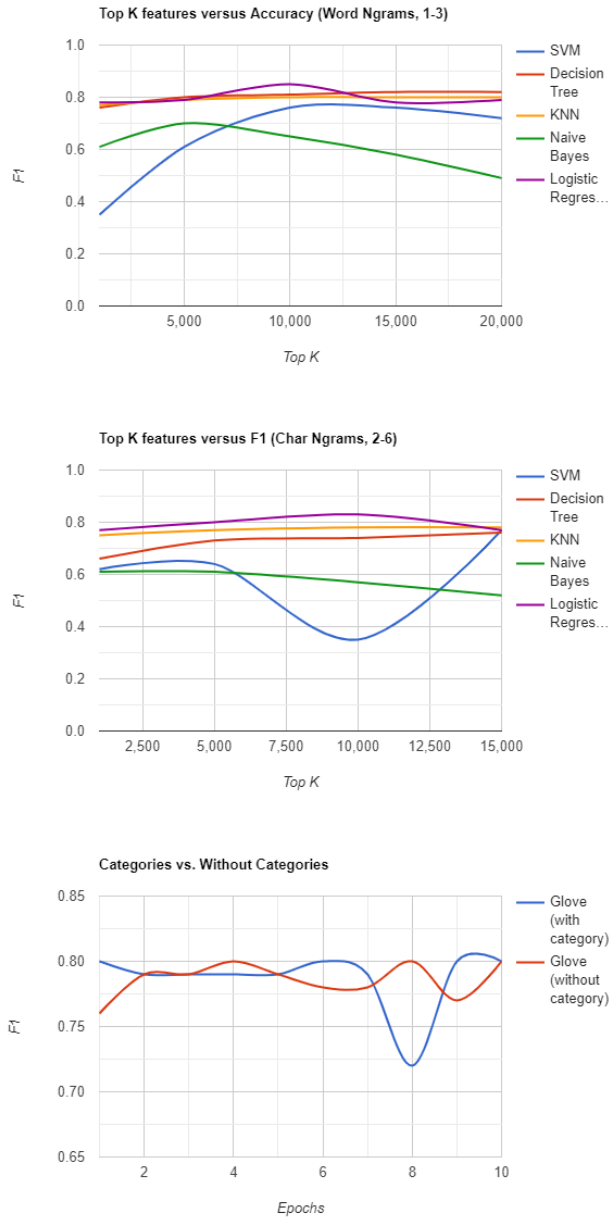

Figure 5: First two line charts illustrate the variations in F1 score by simple classification models after varying the number of features. The first line chart depicts the results of word ngrams whereas the second one shows the results for character ngrams. The last line graph presents comparison between Glove embeddings (with and without category feature).

REFERENCES

[1] Sara Abdollahi, Simon Gottschalk, and Elena Demidova. 2020. Eventkg+ click: a dataset of language-specific event-centric user interaction traces. *arXiv preprint arXiv:2010.12370*.

[2] Hosam Al-Samarraie, Atef Eldenfria, and Husameddin Dawoud. 2017. The impact of personality traits on users' information-seeking behavior. *Information Processing & Management*, 53, 1, 237–247.

[3] Tsan-Kuo Chang and Jae-Won Lee. 1992. Factors affecting gatekeepers' selection of foreign news: a national survey of newspaper editors. *Journalism Quarterly*, 69, 3, 554–561.

[4] Verena Eitle and Peter Buxmann. 2020. Cultural differences in machine learning adoption: an international comparison between germany and the united states.

[5] Meihan He and Jongsu Lee. 2020. Social culture and innovation diffusion: a theoretically founded agent-based model. *Journal of Evolutionary Economics*, 1–41.

[6] Mahmood Khosrowjerdi, Anneli Sundqvist, and Katriina Byström. 2020. Cultural patterns of information source use: a global study of 47 countries. *Journal of the Association for Information Science and Technology*, 71, 6, 711–724.

[7] Gregor Leban, Blaz Fortuna, Janez Brank, and Marko Grobelnik. 2014. Event registry: learning about world events from news. In *Proceedings of the 23rd International Conference on World Wide Web*, 107–110.

[8] Björn Preuss. 2017. Text mining and machine learning to capture cultural data. Technical report. working paper, 2. doi: 10.13140/RG. 2.2. 30937.42080.

[9] Giselle Rampersad and Turki Althiyabi. 2020. Fake news: acceptance by demographics and culture on social media. *Journal of Information Technology & Politics*, 17, 1, 1–11.

[10] H Denis Wu. 2007. A brave new world for international news? exploring the determinants of the coverage of foreign news on us websites. *International Communication Gazette*, 69, 6, 539–551.

Zotero to Elexifinder: Collection, curation, and migration of bibliographical data

David Lindemann
david.lindemann@ijs.si
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia

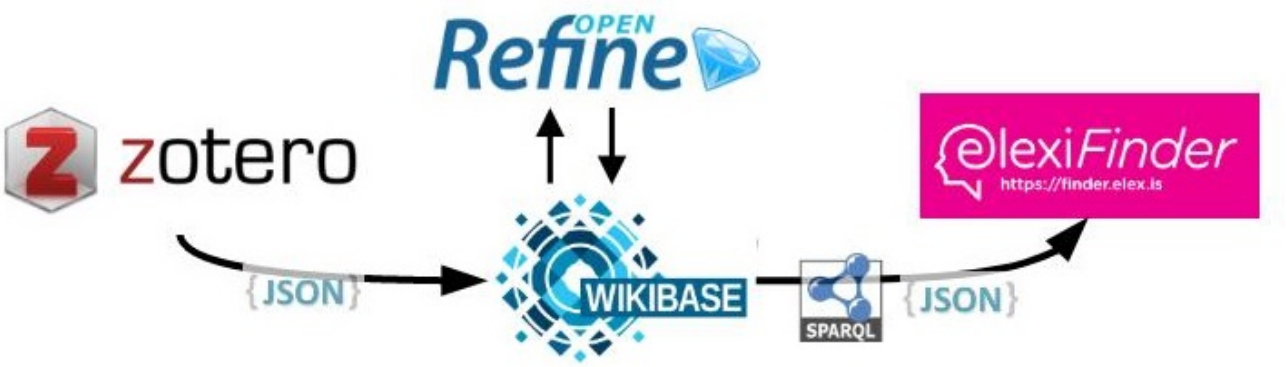

Figure 1: Zotero to Elexifinder workflow model

ABSTRACT

In this paper, we present ongoing work concerning a workflow and software tool pipeline for collecting and curating bibliographical data of the domain of Lexicography and Dictionary Research, and data export in a custom JSON format as required by the Elexifinder application, a discovery portal for lexicographic literature. We present the employed software tools, which are all freely available and open source. A Wikibase instance has been chosen as central data repository. We also present requirements for bibliographical data to be suitable for import into Elexifinder; these include disambiguation of entities like natural persons and natural languages, and a processing of article full texts. Beyond the domain of Lexicography, the described workflow is applicable in general to single-domain small scale digital bibliographies.

KEYWORDS

bibliographical data, author disambiguation, e-science corpora

1 INTRODUCTION

In 2019, version 1 of Elexifinder,[1] a discovery portal for lexicographic literature, was launched in the framework of the ELEXIS project [2].[2] At the same time, at University of Hildesheim, a domain ontology and bibliographical data collection for Lexicography and Dictionary Research was planned [6, 5]. Both endeavours already had compiled significant datasets. At a dedicated workshop connected to the 2019 eLex conference in Sintra (Portugal), it was decided to combine the efforts, and the workflow explained in this paper was designed, in order to merge existing datasets, decide criteria for data curation, and make the results available to the lexicographic community. Two years later, at the 2021 Euralex conference, Elexifinder version 2 was introduced [3]. Main shortcomings of Elexifinder version 1 have been sorted out, namely the missing author disambiguation, and the coverage of the domain's literature has been significantly increased, also regarding publication languages other than English. Moreover, a vocabulary of lexicographic terms has been developed, which is now used for content-describing indexation of article full texts.

Lexicography and Dictionary Research is a relatively small discipline, having thematic intersections with Corpus Linguistics, Terminology, Natural Language Processing, and Philology. In metalexicographic literature, all aspects of the lexicographic process, dictionary structure and functions, dictionary use, and other relevant issues are discussed. The lexicographic community communication is mainly taking place through a reduced number of conference series and journals, being complemented by handbooks and other edited volumes. The need for a dedicated digital bibliography arises from the following observations:

- The vast majority of publications do not have Digital Object Identifiers (DOI), and thus are not indexed in cross-domain digital collections of publication metadata. This applies to nearly all older publications, but also to many newer contributions published in the last two decades.
- When searching for metalexicographical publications in cross-domain digital collections, search results are mixed up with publications from other domains, which may disturb a straightforward information retrieval.
- Author disambiguation in domain-independent digital collections that can be considered the big players in the field (such as Google Scholar) is not at all accurate, so that very

[1] Accessible at https://finder.elex.is.
[2] See https://elex.is.

Information Society 2020, 5–9 October 2020, Ljubljana, Slovenia

often name variants are not resolved to a single person entity, and different persons with the same name are not disambiguated.
- If articles are indexed with content-describing terms in cross-domain digital collections, the vast majority of those terms will be out of the scope of the domain we are looking at.
- Publication metadata found at big (i.e. automatically compiled) repositories is often incomplete or noisy, so that using those, e.g. for citations, requires manual intervention in order to achieve a publishable quality.

Therefore, it seems useful to provide the lexicographic community with a platform that makes publications and their metadata accessible in a way that the described shortcomings will be overcome. Single-domain endeavours of this kind, which all involve manual curation, are *DBLP*[3] for Computer Science, *IxTheo*[4] for Theology, or *EconBiz*[5] for Economics. Inspired by features found in these, we propose a workflow that involves the use of free software accessible to anybody, which makes it reproducible and cost-reducing.

2 LEXBIB ZOTERO GROUP

Zotero,[6] developed and maintained by the Corporation for Digital Scholarship[7], a non-profit organisation, is the most widely used open source citation management software application. Zotero offers functionality for web-scraping publication metadata, importing metadata from different structured formats, and an online platform for collaborative curation of metadata, along with the possibility to attach full text PDF (and TXT versions) to metadata records. The Zotero scraper functionality allows to download publication metadata and attached PDF files from all those sites the Zotero community has provided a "translator"[8] for, including the web platforms of major publishing houses, Open Journal Systems, etc. From the Zotero platform, users are able to obtain metadata records as single items or as batches for import into their own citation managers, or as export records in a range of citation styles or in structured formats such as bibtex. Members of a Zotero group can view and download full text attachments. Moreover, Zotero items can be annotated with custom tags, and additional information (such as excerpts or comments) can be attached to them. Around Zotero, an active community is developing plug-ins that add new functionalities to Zotero.[9]

In the first planning period of the LexBib project, funded by the University of Hildesheim, conference publications of the Euralex and the eLex conference series, and publications from a range of journals and edited volumes have been added to LexBib Zotero group.[10] Items collected for Elexifinder version 1, available as tabular data, have then been merged to the Zotero group. For this purpose, tabular csv data has been transformed to RIS format[11] and imported to Zotero. Additionally, metadata records from OBELEX-meta and EURALEX-Dykstra bibliographies have been added.[12] Duplicate management has been done in batches (whole journal issues or conference iterations), or one by one using Zotero's built-in duplicate detection functionality. Main criterion for the inclusion of metadata records has been the availability of the corresponding full texts. This means a clear preference for Open Access publications; but also other publications have been included, wherever a suitable license agreement allowed access to the text.[13]

Zotero data can be accessed by API,[14] or exported locally using pre-set or custom export scripts. We use an adapted version of the Zotero JSON-CSL exporter, which produces a list of JSON objects containing all metadata fields and their values as literal strings, as well as the location of all local file attachment copies. For statements that cannot be expressed using standard Zotero fields[15], we have used Zotero tags as workaround, following a simple syntax of predicate and object. For example, for asserting that an article is a review article, the tag ":type Review", and so on. Tags in Zotero can be easily copied from one item to others by manual drag-and-drop operations, set via API, and also be included in display styles, so that in the Zotero item listings, for example, review article titles can be preceded by a coloured symbol. With this workaround we can assert semantic triples inside Zotero. That is, for instance, that for representing the statement that a certain item is contained in another item (e.g. a book chapter item in an item of type book), we use a tag beginning with ":container", followed by an identifier for the containing item; for a conference paper presented at a certain event, we use a tag beginning with ":event", followed by an identifier for that event. For both of these, corresponding Zotero fields do exist ("contained in", "presented at"), but these are filled by the web scraping and importer translators with literal string values as needed for citations, and not with unambiguous identifiers.

For Elexifinder, a special metadatum is included in all publication metadata sets: The location of the first author. This allows the generation of location maps and search filters according to locations in the Elexifinder portal. For these locations, we insert English Wikipedia page titles in the Zotero "extra" field.[16]

3 LEXBIB WIKIBASE

3.1 Wikibase as LOD infrastructure solution

The decisive shift from a metadata set as in Zotero, which consists of certain fields and their literal values, towards unambiguous Linked Data lies in the reconciliation of those literal values against existing or new unambiguous identifiers. For example, and this already refers to the hardest nut to crack in this context, an author may have several name variants appearing across the publication metadata collection, and there may be other persons sharing the same name, or any of the name variants. But one author or editor (i.e., a "creator") should only have one identifier (such as ORCID). Since we do not know Wikidata and/or ORCID identifiers of all creators in our database, we need to create our own (and map them later). Other Zotero fields that should be

[3] Accessible at https://dblp.org/.
[4] Accessible at https://ixtheo.de/.
[5] Accessible at https://www.econbiz.de/.
[6] See https://zotero.org.
[7] See https://digitalscholar.org/.
[8] See https://www.zotero.org/support/translators.
[9] For example, very recently the Cita plug-in has been developed, which allows to add citation metadata to Zotero records, see https://meta.m.wikimedia.org/wiki/Wikicite/grant/WikiCite_addon_for_Zotero_with_citation_graph_support.
[10] Last version accessible at https://www.zotero.org/groups/lexbib/library.
[11] See https://en.wikipedia.org/wiki/RIS_(file_format).
[12] See references in [3].
[13] Article full text are stored and exclusively used for project-related text mining tasks; they cannot be downloaded from Zotero. We instead provide download links which lead to the download offered by the corresponding publisher, subject to applicable restrictions.
[14] See https://www.zotero.org/support/dev/web_api/v3/start.
[15] See https://www.zotero.org/support/kb/item_types_and_fields.
[16] Wikipedia page titles are unambiguous (see e.g. https://en.wikipedia.org/wiki/Cambridge vs. https://en.wikipedia.org/wiki/Cambridge,_Massachusetts), and map to only one Wikidata entity. This strategy has turned out effective, since manual annotators are able to find the adequate Wikipedia page without hassle.

reconciled against unambiguous identifiers are those describing the containing item, the conference where the contribution was presented, the journal, the publisher, the publication place, and the publication language. For some of these, persistent identifiers are available in many cases (e.g. journals), or in all cases (languages). In general, we create our own identifiers, and map them to Wikidata; in some cases, immediately (languages, places, and, by ISSN, also journals), and in other cases, we leave that mapping to the (near) future, as it is the case for creators and publishers. Other Zotero fields contain identifiers (ISSN, ISBN, DOI), which after normalisation can be taken directly as external identifiers in a Linked Database.

After experimenting with different RDF database solutions, which allow to represent data in the described way, we have decided for Wikibase,[17] which is the software infrastructure underlying main Wikidata.[18] Since 2019, "Wikibase as a Service" is offered to the community.[19] Wikibase entities are items (each of which has its own identifier preceded by the letter Q), and properties (preceded by letter P), just as in Wikidata, but in a different namespace. Properties may point to other items, other properties, external identifiers, or values of a certain datatype, such as "monolingual text", "point in time", "string", "url", etc.[20]

Wikibase as central data repository solution has several advantages compared to other infrastructure solutions for Linked Open Data (LOD):

- Entity data is displayed on entity pages, where it can be viewed and edited. These pages always reflect the last update.
- A complete edit history is available, and changes can be undone.
- Every entity page is linked to a dedicated discussion page.
- User and user rights management allow a community-driven editing process.
- In addition to query interface and SPARQL endpoint known from other RDF database solutions, Wikibase data can be uploaded and downloaded using an API, and as entity data dump in several formats.

The backbone of LexBib Wikibase is an ontology of classes and properties,[21] which can be aligned to Wikidata or other external ontologies. We have started to define these alignments. This ensures interoperability with other resources, such as Wikidata, so that data can be transferred from LexBib to Wikidata or vice versa, or accessed in both at the same time, using federated SPARQL queries.

3.2 Zotero to Wikibase migration

As mentioned before, Zotero item data is exported from a local Zotero instance, using an adapted version of the Zotero JSON-CSL exporter.[22] The resulting list of JSON objects is then processed in the following way:

- Zotero tags that contain semantic triple shortcodes (explained above) are mapped to the corresponding LexBib wikibase properties, in this case with datatype "item", that is, to object properties.
- Creator name and publisher name literals are mapped to the properties corresponding to the creator role (author or editor), or to the publisher. This is done in a way that the name literals appear as qualifiers to a wikibase "novalue" statement, which is a placeholder for the creator or publisher item, that will be defined in the disambiguation process explained below.
- Zotero fields that contain external identifiers (ISSN, ISBN and DOI), are mapped to the corresponding properties of datatype "external identifier". Wikibase properties of that datatype allow to define a URL pattern, in order to make the identifier a valid hyperlink, which can be clicked on in Wikibase entity data pages.
- As mentioned, we use the Zotero "extra" field ("note" in bibtex) for annotation of the item with a Wikipedia page that corresponds to the first author's location. Wikidata API is queried for the corresponding Wikidata entity, an equivalent of which is created in LexBib Wikibase, in order to function as object to the property "first author location".
- The Zotero "language" field, in LexBib may contain a two-letter ISO-639-1, or a three-letter ISO-639-3 code. This is mapped to a property pointing to the language item corresponding to that code.
- The Zotero item URI is taken as external identifier in LexBib wikibase, with the Zotero storage location of PDF and TXT attachments as qualifiers to that statement. In addition, we annotate this statement with a qualifier asserting the presence of an abstract, and, if any, in what language.[23]
- The content of the remaining fields is mapped to Wikibase properties of the corresponding datatype ("URL", "string", or "point in time").

The resulting dataset is then imported into LexBib Wikibase. It is worth mentioning that uploading data to a Wikibase triple by triple using the mediawiki API of the Wikibase instance[24] takes about 0.5 seconds per triple, which is due to the need of updating Wikibase search indices and edit histories for every single uploaded triple.

3.3 Entity disambiguation using Open Refine

The around 5,000 creator names appearing in LexBib Zotero by spring 2021 have been mapped to around 4,000 unique person items. This has been done testing different clustering algorithms available in the Open Refine application,[25] by Christiane Klaes from the University of Hildesheim, in the framework of her MA thesis [1]. These are the creator items present in LexBib Wikibase experimental version 2.[26]

From that moment on, any new Zotero item that is exported to Wikibase, which will contain, as explained above, one or more creator statements of type "novalue", is reconciled against existing LexBib Wikibase creator items, using the given and last name literal qualifiers. For this purpose, a reconciliation service for LexBib Wikibase is set up[27], and then accessed by Open Refine, in order to match creator name literals to creator items.

[17] See http://wikiba.se; our instance is accessible at http://lexbib.elex.is.
[18] Accessible at http://www.wikidata.org.
[19] See https://www.wbstack.com. The service has been co-enabled by Adam Shoreland (https://addshore.com/), Rhizome (https://rhizome.org/), and WMDE (https://www.wikimedia.de/).
[20] See https://www.wikidata.org/wiki/Help:Data_type.
[21] For more information, see LexBib Wikibase main page at https://lexbib.elex.is.
[22] Available at https://github.com/elexis-eu/elexifinder/blob/master/Zotero/LexBib_JSON.js.
[23] The abstract language is assumed to be the same as the publicaton language, if not stated different as tag shortcode ":abstractLang".
[24] For LexBib Wikibase, see https://lexbib.elex.is/w/api.php.
[25] Available at https://openrefine.org/.
[26] Accessible at https://data.lexbib.org.
[27] This is done using https://github.com/wetneb/openrefine-wikibase.

If a literal can not be matched to any existing item, a new person item is created. The reconciliation also works with fuzzy matches, and all name variants attached to existing items are considered. Matches can also be manually chosen. Any additional name variant appearing in Zotero data is linked to the LexBib Wikibase person item as "alias" label, while the most frequent name variant is chosen as "preferred" label. This allows for the new name variants being available for subsequent reconciliation iterations.

LexBib persons have up to six name variants found in Zotero data. In some cases, we have chosen the preferred name variant manually, according to the author's own choice, or to conventions in the community regarding the naming of commonly known authors.[28]

3.4 Full text processing

LexBib full text PDFs are stored in the local Zotero storage folder, which is automatically synchronised with Zotero cloud. When processing Zotero JSON output, PDF files are sent to an installation of the GROBID application[29], which will propose a TEI representation of the PDF content. This allows for isolating the full text body from the other text components, such as title, running titles, abstract, author list, and references section. The extracted full text body is manually validated, and, in case of any mistake, it is corrected, using a plain TXT version of the PDF, which is by default produced by Zotero.

GROBID turns out to structure PDF content as TEI very efficiently if the article resembles a typical structure as found in journals and proceedings. Book chapters and review articles, which normally do not feature an abstract, in turn, are usually not parsed adequately. In those cases, we now use directly the plain TXT version for producing a cleaned version manually.

The article text is then lemmatised,[30] and lexicalisations of LexVoc lexicographic terms are looked up in the text.[31] LexVoc vocabulary[32] is a resource still under development; for the term discovery process, terms and lexicalisations (labels) are obtained from LexBib Wikibase by a SPARQL query, the result of which will reflect the state of LexVoc in that particular moment. The keyword processor returns counts of every term, so that relative frequencies can be calculated for every term, according to the occurrences of its labels and the amount of tokens in the article text body; this information can be uploaded to LexBib Wikibase bibliographical items, so that term indexation becomes part of their entity data.

4 WIKIBASE TO ELEXIFINDER

The described workflow is necessary for being able to export bibliographical data in a custom JSON format, as needed for Elexifinder, which is an application based on some of the elements of the Event Registry system architecture [4]. In particular, authors and content-describing terms (Elexifinder "categories") have to be represented as objects containing an unambiguous URI and a textual label; the containing item, the LexBib Zotero item URI, and the link for accessing full text download are represented as URL, publication date in ISO 8601 format, publication language in ISO 639-3 format, and the item title as simple string.

The full text body itself is also exported to Elexifinder, where it is used for displaying the first bits of it in search result displays, and for wikification, from which Elexifinder "concepts" are obtained, as long as the system is able to associate named entities occurring in the text with Wikipedia pages that describe them.

5 CONCLUSIONS AND OUTLOOK

The described workflow enables us to disambiguate entities found in bibliographical datasets. For the time being, we are applying this for feeding the Elexifinder app. Having chosen Wikibase as central data repository also allows for aligning LexBib data with Wikidata in a straightforward way. In some cases, we have imported statements from Wikidata, in order to enrich LexBib entities with additional information, but that can be done the other way round as well. In other words: Wherever we find (or create) a Wikidata entity to align with our own, we can export the statements asserted on LexBib Wikibase to the main Wikidata. We have done this using LexBib events (conferences) as test case, and plan to align other entity types with Wikidata in the near future, namely articles, persons, and organisations.

ACKNOWLEDGMENTS

The research received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 731015.

REFERENCES

[1] Christiane Klaes. 2021. *Linked Open Data-Strategien zum Identity Management in einer Fachontologie*. Master's thesis. Universität Hildesheim, Hildesheim, (June 2021). http://lexbib.elex.is/entity/Q15468.

[2] Iztok Kosem and Simon Krek. 2019. ELEXIFINDER: A Tool for Searching Lexicographic Scientific Output. In *Electronic Lexicography in the 21st Century: Proceedings of the eLex 2019 Conference*. Lexical Computing CZ s.r.o., Brno, 506–518. http://lexbib.elex.is/entity/Q9484.

[3] Iztok Kosem and David Lindemann. 2021. New developments in Elexifinder, a discovery portal for lexicographic literature. In *Lexicography for Inclusion: Proceedings of the 19th EURALEX International Congress, 7-11 September 2021, Alexandroupolis, Vol. 2*. Democritus University of Thrace, Alexandroupolis, 759–766. http://lexbib.elex.is/entity/Q15467.

[4] Gregor Leban, Blaz Fortuna, Janez Brank, and Marko Grobelnik. 2014. Event registry: learning about world events from news. In *Proceedings of the 23rd International World Wide Web Conference, WWW14, Seoul, Korea, April 7-11, 2014*, 107–110. DOI: 10.1145/2567948.2577024.

[5] David Lindemann, Christiane Klaes, and Philipp Zumstein. 2019. Metalexicography as Knowledge Graph. *OASICS*, 70. http://lexbib.elex.is/entity/Q13955.

[6] David Lindemann, Fritz Kliche, and Ulrich Heid. 2018. LexBib: A Corpus and Bibliography of Metalexicographical Publications. In *Lexicography in Global Contexts: Proceedings of the 18th EURALEX International Congress, 17-21 July 2018, Ljubljana*. Ljubljana University Press, Ljubljana, 699–712. http://lexbib.elex.is/entity/Q6059.

[28] See an example at http://lexbib.elex.is/entity/Q1583.
[29] See https://grobid.readthedocs.io.
[30] For the time being, we are only processing English text. For lemmatisation, we use spaCy (see https://spacy.io/).
[31] This is done using https://pypi.org/project/flashtext/.
[32] Described at http://lexbib.elex.is/wiki/LexVoc.

Simple Discovery of COVID IS WAR Metaphors Using Word Embeddings

Mojca Brglez
University of Ljubljana
Ljubljana, Slovenia
mojca.brglez@ff.uni-lj.si

Senja Pollak
Jožef Stefan Institute
Ljubljana, Slovenia
senja.pollak@ijs.si

Špela Vintar
University of Ljubljana
Ljubljana, Slovenia
spela.vintar@ff.uni-lj.si

ABSTRACT

In the past year, the discourse on the COVID-19 pandemic has produced a great number of metaphors stemming from the more basic conceptual metaphor ILLNESS IS WAR. In this paper, we present a semi-automatic method to detect linguistic manifestations of the latter in Slovene media. The method consists of assembling a seed vocabulary of war-related words from an existing Slovene metaphor corpus, extending the vocabulary using word embeddings, and refining the extended vocabulary using intersection filtering. Our method offers a quick compilation of corpus data for further analysis, however, we also address issues related to the method's precision and the need for manual filtering.

KEYWORDS

metaphors, covid, word embeddings, media discourse

1 INTRODUCTION

The COVID pandemic has been a ubiquitous topic in the discourse of the past year, featuring in medical, political, public and personal discourse. The emergence of a new virus of yet unknown origin, behaviour and effects has presented itself like a complex and obscure topic. To make sense of it, we have once more resorted to metaphorical language, much like we do when faced with other abstract, obscure concepts. According to Conceptual Metaphor Theory (CMT, [11, 12]), metaphors "are among our principal vehicles for understanding" and "play a central role in the construction of social and political reality" ([12, p. 151]). In CMT, linguistic metaphors such as *"food for thought"* and *"half-baked idea"* are considered manifestations of an established conceptual mapping between a more concrete domain and a more abstract domain, here for example IDEAS ARE FOOD. The domain of DISEASES, on the other hand, is often mapped to the domain of WAR, a more common frame of reference which has taken hold as a fairly conventional way to talk about illnesses and their treatments, as well as several other domains ([8]).

As was already observed in various studies ([19, 2, 5, 7]), the discourse on the current COVID pandemic has also repeatedly used the WAR domain in its metaphors. At the time of our experiment, however, no study has yet addressed the use of such metaphors in Slovene, where they were also adopted for communicating various implications, preventive measures, recommendations and laws to abide by. To investigate the use and pervasiveness of this metaphorical domain in Slovene media, we have conducted a quick analysis of a corpus of COVID-related news articles using an innovative methodological approach. We propose a top-down method to search for expected conceptual metaphors through semi-automatic means employing word embeddings. While most previous corpus-based approaches to identify metaphors either use a small set of candidate words or require manual inspections of large data samples, our approach reduces manual work on assembling linguistic data by combining existing annotated resources and text mining methods.

Information Society 2021, 4–8 October 2021, Ljubljana, Slovenia

2 PROPOSED APPROACH

Our method aims to discover linguistic expressions of the conceptual metaphor COVID IS WAR in the corpus by targeting a broader potentially metaphoric vocabulary. Previous related works have relied on either a limited vocabulary set (e.g. [7]) or a list of words laboriously compiled from various sources such as dictionaries, thesauri and other studies on metaphor [19], or have used sophisticated but complex NLP methods and specialized resources (e.g. [6]). In our experiment, we use a simple unsupervised approach using existing resources and language processing technologies.

The main novelty of our approach is using pre-trained word embeddings to extend the vocabulary, used also by e.g. [16] and [18] to extend terminology. As past research has shown [14], word embeddings used for training language models retain linguistic regularities, including syntactic and semantic relationships between words. This means that similar words have similar vectors, and the closer vector representations (word embeddings) are, the higher the chance they share a certain semantic space. We make use of this feature by trying to capture a semantic space that would resemble the conceptual domain of WAR, which represents the source domain of the metaphor.

2.1 Method

First, we start by collecting war-related lexical units from the KOMET corpus [1], the only corpus of metaphors in Slovene which was recently compiled and annotated similarly to the English corpus of metaphors, VUAMC [17]. KOMET contains approximately 200,000 words obtained from journalistic, fiction and online texts and was hand-annotated for metaphoricity on the basis of the MIPVU procedure ([17]). Additionally, the metaphoric expressions are tagged for one of 69 semantic frames, i.e. the source concepts that semantically motivate them. One of these semantic frames is `#met.battle`, which subsumes 105 metaphoric instances with 67 different lemmas, such as *predati, ostrostrelec, orožje, napasti* [surrender, sniper, weapon, attack]. These also form multi-word idioms such as *železna pest* [iron fist] and *boriti se z mlini na veter* [to tilt at windmills] which we exclude from our candidates list because the word embeddings we use only represent tokens, not whole phrases. Moreover, the lemmas within do not themselves necessarily represent the desired domain. We also filter out some words erroneously annotated with the frame such as *številen* [numerous]. This gives a starting vocabulary of

51 unique seed words. Then, to extend the vocabulary further, we employ Slovene word token embeddings ([13] pre-trained with fastText ([4]) on various large corpora of Slovene (GigaFida, Janes, KAS, slWaC etc.). For each seed word in the list of words extracted from the KOMET corpus, we use the Gensim library ([20]) to find the word's N nearest neighbours in the fastText embeddings' space (using the `most_similar` function).

To increase the robustness of the extended vocabulary, we try to automatically filter out lexis not related to war. To this end, we use the word embeddings intersection method ([18]). The method retains only the candidates that intersect between the sets, meaning they occur in the neighbourhood of at least k input seed words. For our main experiment, presented in this paper, we select the parameters N=50 and k=3. We thus obtain a maximum of 2550 (50 x 51) potential candidates. In the output, there are 2078 unique words, and, after lemmatization, 1539 unique lemmas. After the intersection filtering, the vocabulary extended by word embeddings consists of 184 word lemmas: 44 of them are already included in our initial seed set and 140 are new lemmas. We join the new, extended set with the initial seed set, which yields a total of 191 lemmas to search for.

3 CORPUS

The experiment is carried out on a corpus of Slovene COVID-19-related news articles, automatically crawled from the web by searching for the keyword "covid-19" in article titles (a subset of the Slovene corpus used in the Slav-NER 2021 shared task ([15]). The corpus consists of 233 texts spanning from February 2nd to December 11th, 2020 . To prepare it for analysis, we remove the header of each text (comprised of the article number, locale, date and URL), then parse the text into sentences and tokens using the NLTK library ([3]). We also lemmatize the corpus using the LemmaGen lemmatization module ([9]). The pre-processed corpus contains 7,273 sentences and 151,947 tokens.

3.1 Corpus search

In the next step, we extract all sentences from the corpus containing any of the war-related terms from our expanded vocabulary of 191 lemmas. The results yield 335 instances of potentially metaphorical expressions. Out of the 191 lemmas on the metaphorical candidate list, the COVID corpus contains 49, appearing in 268 sentences. Due to the unsupervised approach these are still only candidate words from the semantic domain of war. A manual analysis shows that in addition to war metaphors, our extracted sentences include the following four cases:

(1) Some of the seed words found in the corpus are used literally;
(2) Some of the seed words found in the corpus are a result of lemmatization errors
(3) Some of the seed words found in the corpus are used metaphorically, but refer to other target domains, such as POLITICS or NATURE (e. g. *boriti se proti podnebnim spremembam* ['fight against climate change'])
(4) Some of the seed words in our initial 191-candidate list are not actually related to the topic of WAR but are more closely related to another topic (e.g. *gol* ['goal'])

On this account we perform a manual analysis of the extracted sentences and categorize them as follows:

(1) falsely extracted instances due to a lemmatization error or literal use, or true metaphorical expressions but with other source or target domain, and
(2) true metaphorical expressions referring to disease as target domain

For example, in the following sentence, the word brigade [brigades] only refers to a name of a street, which we mark as literal usage.

- */…/ odvzem brisov pri pacientih s sumom na Covid-19: ob Cesti proletarskih **brigad** 21 /…/*
  */…/ taking swabs from patients with suspected Covid-19: at 21, Proletarian **Brigades** Road /…/*

In the following example, the word napad [attack] is used to refer to another domain – INTERNET, COMPUTING, which we mark as metaphor for another target domain.

- *Covid-19 je okrepil trend rasti kibernetskih **napadov** [Covid-19 reinforced the growing trend of cyber **attacks**]*

The following three example sentences contain expression that we mark as metaphor for the target domain of DISEASE.

- *Čeprav v **boju** z virusom to nikakor ni hitro.*
  *[Although this is by no means fast in the **fight** against the virus.]*

- *Kako bo jeseni, ko bodo »**udarili**« še drugi virusi?*
  *[What will happen in autumn, when other viruses also **"strike"**?]*

- *Prvi organski sistem v organizmu, ki ga virus **napade**, povzroči pljučnico, …*
  *[The first system in the organism that the virus **attacks** causes pneumonia …]*

Results of this analysis are presented in Table 1, whereby we report only lemmas that were metaphorically used for the DISEASE target domain at least once.

As can be derived from Table 1, our proposed method correctly identified 25 different lemmas with a total of 123 occurrences that are used metaphorically to frame the topic of the pandemic. Out of our 233 articles, 68 or 29,18% contained at least one militaristic metaphorical expression. The ostensibly most frequent expression used was *boj* [fight] with 46 metaphorical occurrences, followed by *boriti* [to fight] with 13 metaphorical occurrences and *soočati* [to confront] with 7 metaphorical occurrences. They account for 37.4%, 10.6% and 5.7% of all metaphorical expressions found by our method, respectively, and together, they represent more than 50% of them. This points to the interpretation that the news corpus contains mostly highly conventional and recurrent metaphors. A lot of the war-related vocabulary (potential candidates in our extended war-related lexis) is not used, meaning the corpus does not, at this moment, exhibit very original, novel metaphorical expressions. Using a larger and a more recently compiled corpus would perhaps reveal a more innovative use of COVID IS WAR metaphors. The vocabulary extension method using word embeddings has proven fruitful as it revealed some metaphorical expressions that were not in the initial 51-word list extracted from the KOMET corpus. The 9 newly discovered lemmas are: *soočiti, izbojevati, zmagati, obraniti, uiti, soočanje, spopadati, zoperstaviti, podleči* [to confront, to fight, to win, to defend, to escape, confrontation, to combat, to oppose, to succumb].

The analysis also revealed some additional lemmas that relate the epidemic to the war frame. In the sentences containing the lemmas we searched for, there were other words from the WAR

Table 1: Analysis of metaphoric lemmas from the extended vocabulary

Lemma	Corpus occurences	Literal uses, lemmatization errors or other source/target domain	DISEASE as target domain
Boj [fight]	57	11	46
Boriti [to fight]	16	3	13
Soočati [to confront]	17	10	7
Spopad [to combat]	6		6
Spopadanje [combatting]	6		6
Zoperstaviti [to oppose]	5		5
Bitka [battle]	5	1	4
Napad [attack]	41	37	4
Podleči [succumb]	5	1	4
Spopadati [to combat]	5	1	4
Bojen [combat [ADJ]]	17	15	2
Borba [battle]	3	1	2
Braniti [to defend]	4	2	2
Napasti [to attack]	6	4	2
Obramben [defense [ADJ]]	9	7	2
Soočanje [confronting]	2		2
Soočiti [to confront]	6	4	2
Žrtev [victim]	49	47	2
Borec [fighter]	3	2	1
Izbojevati [to fight]	1		1
Obraniti [to defend]	1		1
Štab [base, headquarters]	3	2	1
Udariti [to hit]	2		2
Uiti [to escape]	2	1	1
Zmagati [to win]	5	4	1
TOTAL	270	147	123

domain forming so called metaphor clusters ([10]). Thus, we managed to capture some metaphorical expressions that appeared in close vicinity (in the same sentence) of the found metaphorical expressions: *fronta, strategija, preboj, akcijski načrt, vojna mentaliteta, sovražnik* [front, strategy, breakthrough, action plan, war mentality, enemy]. For instance, our method found the sentence below which, in addition to the word *bitka* [battle] in our candidate list, contains a metaphorical use of the word *fronta* [front].

- ***Bitka*** *proti virusu na več* ***frontah***
  [Battle *against the virus on multiple* ***fronts]***

4 ANALYSING DIFFERENT PARAMETER SETTINGS

Some of the expressions mentioned above would have been captured had we modified the parameters of vocabulary extension. Namely, we experimented with using more nearest neighbours (75, 100, 150 and 200). Our initial experiments were carried out on a N of 50 and intersection k of 3. However, by changing the parameters, the results of initial new lemmas could differ. In Figure 1, we analyse how the seed list changes with different parameters: N of 50 and 75 neighbours, each combined with the intersection count k of 2, 3 and 4. Note that these refer only to the list of potentially metaphoric lemmas, and not to the analysis of their use, which can only be analysed in context. We see that the initially selected parameters (50 neighbours and 3 recurrences) are an acceptable middle-ground between precision and size while still maintaining an unsupervised approach, however, had we wanted more examples, we could increase the parameter N or decrease the parameter k.

For the recall, we are not able to carry out a systematic evaluation. Nevertheless, based on metaphor clusters analysis mentioned above, we identified the set of additional words that belong to the military vocabulary: *fronta, strategija, preboj, akcijski, vojen, sovražnik* [front, strategy, breakthrough, action [ADJ], war [ADJ], enemy]. The words *vojen* [war[ADJ]] and *sovražnik* [enemy] would have been included if we lowered the intersection parameter to $k = 2$ at $N = 50$ neighbours or extended the vocabulary by $N = 75$ neighbours while keeping the intersection parameter $k = 3$. Other metaphorical expressions occurring in the corpus *(fronta, preboj, strategija, akcijski)* [front, strategy, breakthrough, action [ADJ]] are not found anywhere in the first 200 neighbours of any of the words, indicating perhaps that the number of neighbours might be further increased. However, we observe that increasing the number of neighbours leads to fuzzier results. The added vocabulary using 75, 100, 150, and 200 nearest neighbours of our initial seed words includes increasingly more words unrelated to the topic of war and some very common words, which would need additional filtering. We assume that the reason for this is that words commonly used metaphorically (conventional or dead metaphors) are "displaced" in the vector space of embeddings, moving away from the words in their original semantic domains and closer to words in other semantic domains – target domains. For example, we observed a lot of sports expressions in our extended vocabulary (e.g. "ball", "goal", "goalpost"). This shows how entrenched metaphors are in our language: in the vector space of word embeddings, the semantic domains are already "muddled". In the present example, this could be a due to the frequent linguistic manifestations of the conceptual metaphor COMPETITION IS WAR.

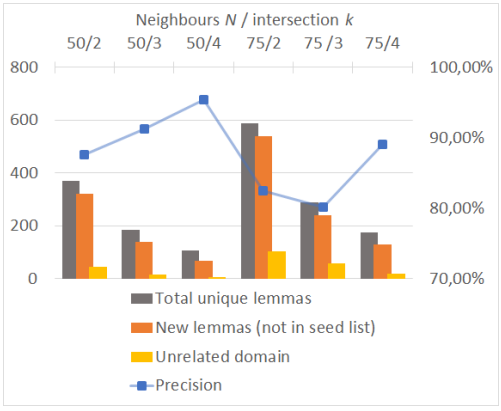

Figure 1: Analysis of vocabulary extension parameters *N* and *k*

5 CONCLUSION

We present an innovative approach using word embeddings as a tool for extending the vocabulary of potentially metaphoric expressions and identify them in corpora. Our approach shows promise in that it correctly identifies numerous such expressions and confirms that intersections of semantic spaces of metaphorical seed words can be used to refine the quest for words pertaining to the military domain. Nevertheless, some metaphoric expressions are missed by our method and the experiment still needs manual analysis. Further research and experiments would be needed for a larger expansion of vocabulary and a finer filtering approach as well as comparing different word embeddings, possibly those trained on more literal language.

ACKNOWLEDGMENTS

This work is supported by the Slovenian Research Agency by the research core funding P6-0215 and P2-0103, as well as by the research project CANDAS (J6-2581). The work has also been supported by the European Union's Horizon 2020 research and innovation programme under grant agreement No. 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media). The results of this paper reflect only the authors' view and the Commission is not responsible for any use that may be made of the information it contains.

REFERENCES

[1] Špela Antloga. 2020. Metaphor corpus KOMET 1.0. Slovenian language resource repository CLARIN.SI. (2020). http://hdl.handle.net/11356/1293.

[2] Benjamin R. Bates. 2020. The (in)appropriateness of the war metaphor in response to SARS-CoV-2: a rapid analysis of Donald J. Trump's rhetoric. *Frontiers in Communication*, 5, 50, (June 2020). DOI: 10.3389/fcomm.2020.000505.

[3] Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python*. (1st edition). O'Reilly Media, Inc.

[4] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5, (June 2017), 135–146. DOI: doi.org10.1162/tacl_a_00051.

[5] Eunice Castro Seixas. 2021. War metaphors in political communication on Covid-19. *Frontiers in Sociology*, 5, 112. DOI: 10.3389/fsoc.2020.583680.

[6] Jane Demmen, Elena Semino, Zsófia Demjén, Veronika Koller, Andrew Hardie, Paul Rayson, and Sheila Payne. 2015. A computer-assisted study of the use of violence metaphors for cancer and end of life by patients, family carers and health professionals. *International Journal of Corpus Linguistics*, 20, 2, 205–231. DOI: 10.1075/ijcl.20.2.03dem.

[7] Damián Fernández-Pedemonte, Felicitas Casillo, and Ana Jorge-Artigau. 2021. Communicating COVID-19: metaphors we "survive" by. *Tripodos*, 2, (February 2021), 145–160. DOI: 10.51698/tripodos.2020.47p145-160.

[8] Stephen J. Flusberg, Teenie Matlock, and Paul H. Thibodeau. 2018. War metaphors in public discourse. *Metaphor and Symbol*, 33, 1, 1–18. DOI: 10.1080/10926488.2018.1407992.

[9] Matjaž Juršič, Igor Mozetič, Tomaž Erjavec, and Nada Lavrač. 2010. Lemmagen: multilingual lemmatisation with induced ripple-down rules. *Journal of Universal Computer Science*, 16, 9, 1190–1214. http://www.jucs.org/jucs_16_9/lemma_gen_multilingual_lemmatisation|.

[10] Veronika Koller. 2003. Metaphor clusters, metaphor chains: analyzing the multifunctionality of metaphor in text. In volume 5, 115–134.

[11] George Lakoff and Mark Johnson. 1980. *Metaphors we live by*. University of Chicago press.

[12] George Lakoff and Mark Johnson. 2003. *Metaphors we live by*. University of Chicago press.

[13] Nikola Ljubešić and Tomaž Erjavec. 2018. Word embeddings CLARIN.SI-embed.sl 1.0. Slovenian language resource repository CLARIN.SI. (2018). http://hdl.handle.net/11356/1204.

[14] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, Atlanta, Georgia, (June 2013), 746–751. https://aclanthology.org/N13-1090.

[15] Jakub Piskorski, Bogdan Babych, Zara Kancheva, Olga Kanishcheva, Maria Lebedeva, Michał Marcińczuk, Preslav Nakov, Petya Osenova, Lidia Pivovarova, Senja Pollak, Pavel Přibáň, Ivaylo Radev, Marko Robnik-Sikonja, Vasyl Starko, Josef Steinberger, and Roman Yangarber. 2021. Slav-NER: the 3rd cross-lingual challenge on recognition, normalization, classification, and linking of named entities across Slavic languages. In *Proceedings of the 8th Workshop on Balto-Slavic Natural Language Processing*. Association for Computational Linguistics, Kiyv, Ukraine, 122–133. https://aclanthology.org/2021.bsnlp-1.15.

[16] Senja Pollak, Andraž Repar, Matej Martinc, and Vid Podpečan. 2019. Karst exploration: extracting terms and definitions from karst domain corpus. In *Proceedings of eLex 2019*, 934–956.

[17] G. Steen. 2010. *A Method for Linguistic Metaphor Identification: From MIP to MIPVU. Converging evidence in language and communication research*. John Benjamins Publishing Company. DOI: 10.1075/celcr.14.

[18] Špela Vintar, Larisa Grcic, Matej Martinc, Senja Pollak, and Uroš Stepišnik. 2020. Mining semantic relations from comparable corpora through intersections of word embeddings. In (May 2020). https://aclanthology.org/2020.bucc-1.5.pdf.

[19] Philipp Wicke and Marianna M. Bolognesi. 2020. Framing COVID-19: how we conceptualize and discuss the pandemic on Twitter. *PLOS ONE*, 15, 9, (September 2020), 1–24. DOI: 10.1371/journal.pone.0240010.

[20] Radim Řehůřek and Petr Sojka. 2010. Software framework for topic modelling with large corpora. In (May 2010), 45–50. DOI: 10.13140/2.1.2393.1847.

Topic modelling and sentiment analysis of COVID-19 related news on Croatian Internet portal

Maja Buhin Pandur
Faculty of Organization and Informatics,
University of Zagreb
Varaždin, Croatia
mbuhin@foi.hr

Jasminka Dobša
Faculty of Organization and Informatics,
University of Zagreb
Varaždin, Croatia
jasminka.dobsa@foi.hr

Slobodan Beliga
University of Rijeka, Department of Informatics &
University of Rijeka, Center for Artificial
Intelligence and Cybersecurity
Rijeka, Croatia
sbeliga@uniri.hr

Ana Meštrović
University of Rijeka, Department of Informatics &
University of Rijeka, Center for Artificial
Intelligence and Cybersecurity
Rijeka, Croatia
amestorovic@uniri.hr

ABSTRACT

The research aims to identify topics and sentiments related to the COVID-19 pandemic in Croatian online news media. For analysis, we used news related to the COVID-19 pandemic from the Croatian portal *Tportal.hr* published from 1st January 2020 to 19th February 2021. Topic modelling was conducted by using the LDA method, while dominant emotions and sentiments related to extracted topics were identified by National Research Council Canada (NRC) word-emotion lexicon created originally for English and translated into Croatian, among other languages. We believe that the results of this research will enable a better understanding of the crisis communication in the Croatian media related to the COVID-19 pandemic.

KEYWORDS

News media, sentiment, emotions, pandemic, lexicon approach, Latent Dirichlet Allocation

1 INTRODUCTION

There are three major approaches to sentiment and emotions analysis in text: lexicon based, machine learning based approach [12] and the most recent deep-learning approach. In this research, we used a hybrid approach by applying the method of Latent Dirichlet Allocation (LDA) for topic modelling [6] and lexicon approach by using NRC word-emotion lexicon [13] for detection of sentiments (positive or negative) and basic emotions, according to Pluchik's model of emotions [15], in extracted topics.

The main goal of this paper is to analyse sentiments and emotions in crises communication in the news related to the COVID-19 pandemic published on the Croatian online portal. Our goal was aggravated in this research because articles belong rather to objective than to subjective type of reporting. Another problem is the lack of lexical resources for sentiment and emotions in the Croatian language. Glavaš and co-workers [10] developed a Croatian sentiment lexicon called CroSentiLex, which consists of positive and negative lists of words ranked with PageRank scores. Nevertheless, there is no available lexicon for the analysis of emotions for the Croatian language. Our analysis uses the NRC word-emotion lexicon, initially developed for English and translated into 104 languages, including Croatian. Such an approach has disadvantages due to cultural differences, but developing emotion lexicons for low-resource languages as Croatian is very demanding. Sentiment analysis of COVID-19 related texts is conducted mainly for texts written in English, such as research by Shofiya and Abidi [17], where the SentiStrength tool was used to detect the polarity of tweets, and support vector machine (SVM) algorithm was employed for sentiment classification. In [14], tweets about COVID-19 in Brazil written in Brazilian Portuguese due to lack of language resources are analysed by translating original text from Portuguese to English and using available resources for English.

Regarding Croatian social media space, Twitter social network communication was analysed through sentiment analysis [2] and COVID-19 information spreading [3]. Crisis communication of Croatian online portals was already explored by topic modelling of COVID-19 related articles [7]. However, in that research, it is not included further sentiment and emotional analysis of topics. In [4], information monitoring and name entity

Information Society 2020, 5–9 October 2020, Ljubljana, Slovenia

recognition were conducted on news portal texts related to pandemics.

2 METHODS

2.1 Latent Dirichlet Allocation

LDA is a generative, probabilistic hierarchical Bayesian model that induces topics from a document collection [5,6]. The intuition behind topic modelling using LDA is that documents exhibit multiple topics. The topic is formally defined as a distribution over fixed vocabulary. Induction of topics is done in three steps:

- Each document in the collection is distributed over topics that are sampled using Dirichlet distribution.
- Each word in the document is connected with one single topic based on Dirichlet distribution.
- Each topic is defined as a multinomial distribution over words that are assigned to the sampled topics.

Topic modelling by LDA is conducted using *stm* package in R [16].

2.2 Number of topics estimation

Before performing the LDA topic modelling, it has to be estimated the number of topics. In this research we used four metrics from the R package *ldatuning*: Arun2010 [1], CaoJuan2009 [8], Deveaud2014 [9], and Griffiths2004 [11]. Measures Arun2010 and CaoJuan2009 have to be minimised, while measures Deveaud2014 and Griffiths2004 have to be maximised. However, as measures, Arun2010 and CaoJuan2009 generally decrease with the number of topics, and measures Deveaud2014 and Griffiths2004 increase with the number of topics, we will choose the number of topics as the value when observed measures start to stagnate.

2.3 Detection of sentiments and emotions

For the association of sentiments and emotions to extracted topics it was used NRC word-emotion lexicon [13], which consists of 14,182 words with scores of 0 or 1, according to the association to *positive* or *negative* sentiment or one of eight emotions of Pluchick's model (*anger*, *anticipation*, *disgust*, *fear*, *joy*, *sadness*, *surprise*, and *trust*) [15]. The lexicon was created manually by crowdsourcing on Mechanical Turk.

For every sentiment and emotion, we created a vector with a distribution of zeros and ones over the words of a controlled dictionary created from the collection. Association of topics to sentiments and emotions is calculated as the cosine similarity between vectors of topics and corresponding vector of sentiment or emotion.

3 EXPERIMENT

3.1 Data set and preprocessing

The data set used for research consists of articles from the Internet portal *Tportal.hr* related to the topics of COVID-19 pandemic crises and collected from 1st January 2020 to 19th February 2021. Each article included in the dataset is defined as a COVID-19 article only if it contains at least one keyword related to coronavirus thematic. We use COVID-19 thesaurus for article filtering, which contains about thirty of the most important words describing the SARS-CoV-2 virus epidemic together with their corresponding morphological variations. From the total of 31,177 articles, according to defined filtering, the dataset used in the experiment consists of 12,080 COVID-19 related articles. Articles on the portal are categorised into one of nine main categories: *Biznis* (*Business*), *Sport* (*Sport*), *Kultura* (*Culture*), *Tehno* (*Techno*), *Showtime*, *Lifestyle*, *Autozona* (*Autozone*), *Funbox*, and *Vijesti* (*News*) (see Table 1).

Documents of a collection are created using text from the article's subcategory, introduction, main text, and tags. The collection is preprocessed by ejection of English and Croatian stop words and numbers and performing a lemmatisation. It is created a term-document matrix using *tf-idf* weighting scheme. The collection is indexed by terms contained in at least four documents of the collection, and the final list of index terms contained 31,121 terms.

Table 1: Number of articles from dataset categorised into one of nine main categories

Category	Number COVID-19 articles
Business	2,767
Sport	2,008
Culture	894
Techno	101
Showtime	1,352
Lifestyle	1,442
Autozone	124
Funbox	58
News	3,334

3.2 Results

As a first step, the number of topics had to be estimated. Since articles on the portal are categorised into nine main categories, we examined a number of topics from 5 to 15. We chose nine topics since the metrics started to stagnate for a higher number of topics (see Figure 1).

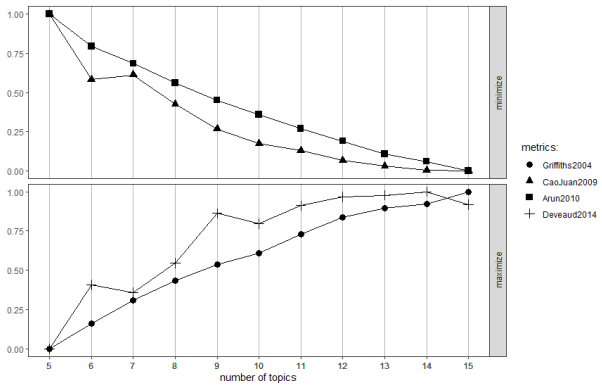

Figure 1: Metrics for estimation of the best fitting number of topics for 5 to 15 topics

Table 2: Top 10 words with the largest probabilities over topics and top 10 words with a *negative* sentiment with the largest probabilities over topics, both sorted in descending order of their probabilities. Topics are sorted by their representation in documents in descending order.

Topic's theme	Top 10 words
Topic 1 – *Sport*	*words by theme:* koronavirus (coronavirus), liga (league), klub (club), nogometni (football), igrač (player), godina (year), utakmica (match), sezona (season), hrvatski (Croatian), nogomet (football)
	words by negative sentiment: igrač (player), velik (big), problem (problem), epidemija (epidemic), odgoditi (to delay), prekinuti (to interrupt), čekati (to wait), borba (fight), napraviti (to make), posljedica (consequence)
Topic 2 – *Vaccination and epidemic measures*	*words by theme:* cijepljenje (vaccination), cjepivo (vaccine), zemlja (country), europski (European), koronavirus (coronavirus), doza (dose), predsjednik (president), vlada (government), mjera (measure), čovjek (man)
	words by negative sentiment: vlada (government), velik (big), epidemija (epidemic), red (order), borba (fight), sud (court), granica (border), problem (problem), potreban (required), upozoriti (to warn)
Topic 3 – *Earthquake and government measures*	*words by theme:* mjera (measure), hrvatska (Croatia), vlada (government), rad (labor), pomoć (help), potpora (support), odluka (decision), potres (earthquake), zaštita (protection), Zagreb
	words by negative sentiment: potres (earthquake), velik (major), pogoditi (to hit), potreban (required), posao (job), šteta (demage), prijava (report), republika (republic), poziv (call), posljedica (consequence)
Topic 4 – *Lifestyle*	*words by theme:* modni (fashion), godina (year), pandemija (pandemic), nov (new), koronavirus (coronavirus), poznat (famous), moda (fashion), obitelj (family), brend (brand), model (model)
	words by negative sentiment: velik (big), nositi (to wear), izolacija (isolation), veza (relationship), majka (mother), dug (debt), djevojka (wench), znak (sign), mali (small), pun (full)
Topic 5 – *Generally stories*	*words by theme:* čovjek (man), vrijeme (time), znati (know), virus (virus), velik (big), život (life), dan (day), dijete (child), koronavirus (coronavirus), dobro (good)
	words by negative sentiment: velik (big), virus (virus), problem (problem), posao (job), napraviti (to make), bolest (disease), mali (small), potreban (required), teško (hard), nositi (to wear)
Topic 6 – *Business 1*	*words by theme:* posto (percentage), godina (year), pad (drop), velik (big), pandemija (pandemic), tržište (market), rast (growth), kuna, gospodarstvo (economy), banka (bank)
	words by negative sentiment: pad (drop), velik (big), kriza (crisis), vlada (government), prihod (income), smanjiti (decrease), mali (small), trošak (expenditure), posljedica (consequence), epidemija (epidemic)
Topic 7 – *Daily reports*	*words by theme:* osoba (person), koronavirus (coronavirus), covid, slučaj (case), mjera (measure), broj (number), županija (county), nov (new), sat (hour), bolnica (hospital)
	words by negative sentiment: bolest (disease), virus (virus), zaraziti (to infect), zaraza (infection), epidemija (epidemic), umrijeti (to die), velik (big), infekcija (infection), zarazan (contagious), simptom (symptom)
Topic 8 – *Culture*	*words by theme:* godina (year), film (film), nov (new), festival (festival), program (program), hrvatski (Croatian), Zagreb, kultura (culture), kazalište (theater), knjiga (book)
	words by negative sentiment: velik (big), mali (small), predstavljati (to present), nastup (appearance), otkazati (to cancel), odgoditi (to delay), smrt (death), rat (war), strana (side), kritika (critique)
Topic 9 – *Business 2*	*words by theme:* nov (new), proizvod (product), automobil (car), velik (big), godina (year), hrvatska (Croatia), proizvodnja (production), tvrtka (company), trgovina (market), kupac (buyer)
	words by negative sentiment: velik (big), nafta (oil), epidemija (epidemic), lanac (chain), smanjiti (decrease), kriza (crisis), mali (small), zaraza (infection), problem (problem), utjecaj (influence)

Topics were labelled based on words with the largest probabilities in topics vectors (keywords) shown in Table 2. Some of the topics are directly connected to main categories on the portal: the first topic is labelled as *Sport*, the fourth topic as *Lifestyle*, and the eighth topic as *Culture*, while the sixth and the ninth topics are connected to the business world and are labelled as *Business 1* and *Business 2*. *Business 1* is associated with the capital market, while *Business 2* is associated with production. Topic 2 is associated with *Vaccination and epidemic measures*, while Topic 3 is associated with *Earthquake and government measures*. Topic 5 seems rather *General on stories* in a pandemic world, while Topics 7 contains *daily reports* on the pandemic state.

We found that all topics are mainly associated with *negative* sentiments. In Table 2 are listed words associated with *negative* sentiment with the largest probabilities across topics, while words associated with *positive* sentiment have coincided with the words from topics theme. This list gives some insight into what "bears" *negative* sentiment in the topics.

Figure 2 shows the association of topics to sentiments and emotions. The ratio of *positive* and *negative* sentiments is the best for categories of *Sport* and *Culture*. These categories and *Lifestyle* are only categories associated with *joy* as one of the dominant emotions. *Surprise* and *anticipation* are dominant emotions across all topics. Categories *Vaccination and epidemic measures*, *Earthquake and government support*, *Generally stories* and *Business 1* are associated with the emotion of *sadness*, while categories *Vaccination and epidemic measures* and *Daily reports* are associated with *fear*.

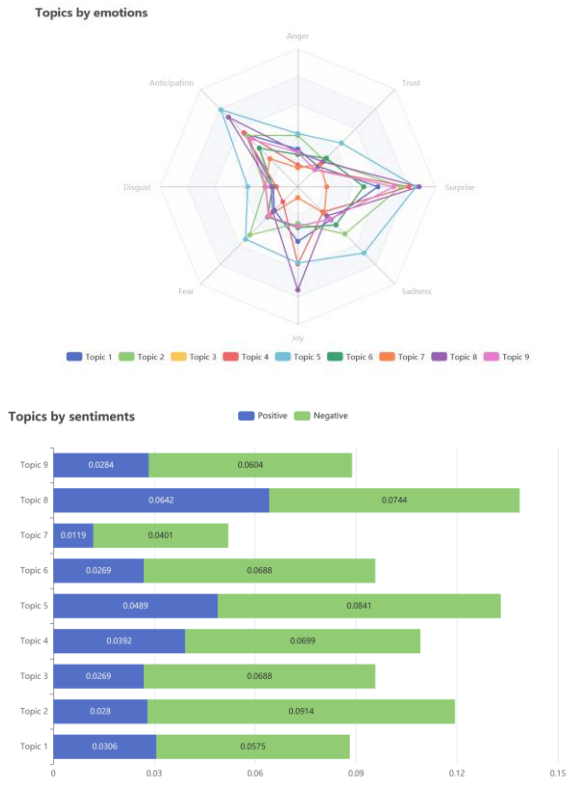

Figure 2: Association of topics to sentiments and emotions

4 CONCLUSIONS AND FURTHER WORK

The main goal of this paper was to analyse sentiments and emotions in crises communication in the news related to the COVID-19 pandemic. For that purpose, we have created our collection of documents from articles on the Internet news portal connected to pandemic crises and analysed it utilising the LDA method for extraction of prevalent topics in the collection and NRC word-emotion lexicon for detection of sentiments and emotions associated with extracted topics.

Application of LDA resulted in relatively intuitive topics. Some of them can be associated with the main categories of the observed portal, and the other are related to the actual situation in a pandemic world in Croatia: *vaccination*, *earthquake* (there were two great earthquakes in Croatia in 2020), *stories*, *daily reports*. It is shown that all extracted topics are associated dominantly with *negative* sentiment, while prevalent emotions are *anticipation*, *surprise*, *sadness* and *fear*.

By this research, we have gained insight into how COVID-19 pandemic crises was communicated to the public. To gain insight into how the public experienced the crises, we could use the same methodology applied to comments of articles or on social networks. This could be a direction for a further work. Also, it would be interesting to investigate how topics and sentiments/emotions are changing and evaluating over time.

ACKNOWLEDGEMENTS

This work has been supported in part by the Croatian Science Foundation under the project IP-CORONA-04-2061, “Multilayer Framework for the Information Spreading Characterization in Social Media during the COVID-19 Crisis” (InfoCoV) and by the University of Rijeka project number uniri-drustv-sp-20-58.

REFERENCES

[1] R. Arun, V. Suresh, C.E. Madhavan and M. Narasima Murty. 2010. On finding the natural number of topics with Latent Dirichlet Allocation: Some observations, In *Proceedings of Advances in Knowledge Discovery and Data Mining, 14th Pacific-Asia Conference* (PAKDD 2010), Hyderabad, India. doi: 10.1007/978-3-642-1357-3_43.

[2] K. Babić, M. Petrović, S. Beliga, S. Martinčić-Ipšić, A. Jarynowski and A. Meštrović. 2022. COVID-19-Related Communication on Twitter: Analysis of the Croatian and Polish Attitudes. In: *Yang XS., Sherratt S., Dey N., Joshi A. (eds) Proceedings of Sixth International Congress on Information and Communication Technology. Lecture Notes in Networks and Systems*, vol 216. Springer, Singapore. Available at https://link.springer.com/chapter/10.1007%2F978-981-16-1781-2_35.

[3] K. Babić, M. Petrović, S. Beliga, S. Martinšić-Ipšić, M. Pranjić and A. Meštrović. 2021. Prediction of COVID-19 related information spreading on Twitter. In *Proceedings of the IEEE International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO 2021)*, accepted for publication.

[4] S. Beliga, S. Martinčić-Ipšić, M. Matešić and A. Meštrović. 2021. Natural Language Processing and Statistic: The First Six Months of the COVID-19 Infodemic in Croatia, In *The Covid-19 Pandemic as a Challenge for Media and Communication Studies*. K. Kopecka-Piech and B. Łódzki, Eds., Routledge, Taylor & Francis Group, accepted for publication.

[5] D. M. Blei. 2012. Probabilistic topic models. *Communications of the ACM*, 55(4), 77-84. doi:10.1145/2133806.2133826.

[6] D. M. Blei, A. Y. Ng and M.I. Jordan. 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research 3*, 993-1022.

[7] P. K. Bogović, S. Beliga , A. Meštrović and S. Martinčić-Ipšić. 2021. Topic modelling of Croatian news during COVID-19 pandemic. In *Proceedings of the IEEE International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO 2021),* accepted for publication.

[8] J. Chao, L. Tian, Z. Jintao, T. Yongdong and S. Tang. 2009. A density-based method for adaptive LDA model selection, *Neurocomputing*, 72(7-9), 1775-1781. doi: 10.1016/j.neucom.2008.06.0011.

[9] R. Deveaud, E. Sanjuan, P. Bellot. 2014. Accurate and effective latent concept modeling for ad hoc information retrieval, *Document Numérique*, 17(1). doi: 10.3166/dn.17.1.61-84.

[10] G. Glavaš, J. Šnajder and B. Dalbelo Bašić. 2012. Semi-supervised acqusition of Croatian sentiment lexicon. In *Proceedings of 15th International Conference on Text, Speech and Dialogue, TSD 2112,* Brno, 166-173.

[11] T.L. Griffiths, M. Steyvers. 2004. Finding scientific topics. In *Proceedings of the National Academy of Sciences 101 Suppl* 1(1), 5228-35, doi: 10.1073/pnas.0307752101.

[12] H. Lane, C. Howard and H. Hapke. 2019. *Natural Language Processing in Action*. Manning Publications, New York, NY.

[13] S. M. Mohammad and P.D. Turney. 2013. Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, 29(3), 436-465.

[14] T. Melo and C. M. S. Figueiredo. 2021. Comparing news articles and tweets about COVID-19 in Brasil: Sentiment analysis and topic modeling approach. *JMIR Public Health and Surveillance*, 7(2), doi: 10.2196/24585.

[15] R. Plutchik. 1962. *The Emotions*. Random House, New York, NY.

[16] M. Roberts, B.M. Stewart and D. Tingley. 2019. stm: An R package for structural topic models, *Journal of Statistical Software*, 91(2), 1-40. doi: 10.18637/jss.v091.i02.

[17] C. Shofiya and S. Abidi. 2021. Sentiment analysis on COVID-19-related social distancing in Canada using Twitter data. *International Journal of Environmental Research and Public Health*, 18(11), 1-10.

Tackling Class Imbalance in Radiomics: the COVID-19 Use Case

Jože M. Rožanec*
Jožef Stefan International Postgraduate School
Ljubljana, Slovenia
joze.rozanec@ijs.si

Tim Poštuvan*
École Polytechnique Fédérale de Lausanne (EPFL)
Lausanne, Switzerland
tim.postuvan@epfl.ch

Blaž Fortuna
Qlector d.o.o.
Ljubljana, Slovenia
blaz.fortuna@qlector.com

Dunja Mladenić
Jožef Stefan Institute
Ljubljana, Slovenia
dunja.mladenic@ijs.si

ABSTRACT

Since the start of the COVID-19 pandemic, much research has been published highlighting how artificial intelligence models can be used to diagnose a COVID-19 infection based on medical images. Given the scarcity of published images, heterogeneous sources, formats, and labels, generative models can be a promising solution for data augmentation. We propose performing data augmentation on the embeddings space, saving computation power and storage. Moreover, we compare different class imbalance mitigation strategies and machine learning models. We find CTGAN data augmentation shows promising results. The best overall performance was obtained with a GBM model trained with focal loss.

CCS CONCEPTS

• **Information systems → Data mining**; • **Computing methodologies → Computer vision problems**; • **Applied computing**;

KEYWORDS

COVID-19, CT Scans, Imbalanced Dataset, Data Augmentation, Computer-Aided Diagnosis, Radiomics, Artificial Intelligence, Machine Learning

ACM Reference Format:
Jože M. Rožanec, Tim Poštuvan, Blaž Fortuna, and Dunja Mladenić. 2021. Tackling Class Imbalance in Radiomics: the COVID-19 Use Case. In *Ljubljana '21: Slovenian KDD Conference on Data Mining and Data Warehouses, October, 2021, Ljubljana, Slovenia.* ACM, New York, NY, USA, 4 pages.

1 INTRODUCTION

In December 2019, an outbreak of the coronavirus SARS-CoV-2 infection (a.k.a COVID-19) began in Wuhan, China. The disease rapidly spread across the world, and on January 30th 2020, the World Health Organization (WHO) declared a global health emergency. The most common COVID-19 symptoms are dry cough, sore throat, fever, loss of taste or smell, diarrhea, myalgia, and dyspnea[5]. In addition, older people, or people with previous medical problems (e.g., diabetes, obesity, or hypertension), are more likely to develop a severe form of the disease[12, 42], which can derive into multiple organ failure, acute respiratory distress syndrome, fulminant pneumonia, heart failure, arrhythmias, or renal failure, among others[37, 40].

Expert radiologists have observed that the impact of the COVID-19 infection on the respiratory system can be discriminated from other viral pneumonia in computed tomography (CT) scans[7, 39]. Most frequent radiological signs include irregular ground-glass opacities and consolidations, observed mostly in the peripheral and basal sites[31]. While such opacities were observed up to a maximum of seven days before the symptoms onset[25], they progress rapidly and remain a long time after the symptoms onset[35, 38]. While such opacities can be observed on chest radiography, they have low sensitivity, which can lead to misleading diagnoses in early COVID-19 stages, and thus a CT scan is preferred[38].

Scientific studies have shown Artificial Intelligence (AI) is a promising technology transforming healthcare and medical practice helping on some clinicians' tasks (e.g., decision support, or providing disease diagnosis)[45]. In particular, the field of radiomics studies how to mine medical imaging data to create models that support or execute such tasks. Given that distinct patterns can be observed on chest radiographies and CT scans, clinicians and researchers sought to use AI for COVID-19 diagnostics[31].

There are multiple challenges associated with radiomics, and in particular, with the COVID-19 diagnosis use case. Despite the limitations that can exist regarding privacy concerns[26, 44], many datasets have been made publicly available. From those datasets, many are limited to a few cases[35]; were collected from different sources and image protocols, and thus cannot be merged (e.g., the gray-levels across images can have different meanings[7]); or were labeled at different granularity levels (e.g., patient-level, or slice-level)[2]. Therefore, models developed from these datasets cannot always be ported to a specific environment. Finally, limitations can exist regarding data collection, further limiting available data to develop working models to diagnose the disease.

The main contributions of this research are (i) a comparative study between four data-augmentation strategies used to deal with class imbalance, (ii) across eight frequently cited machine learning algorithms, based on a real-world dataset of chest CT scans annotated with their COVID-19 diagnosis. We developed the machine learning models with images provided by the Medical Physics

*Both authors contributed equally to this research.

SiKDD '21, October, 2021, Ljubljana, Slovenia

Research Group at the University of Ljubljana and made them available as part of the RIS competition[1].

We report the models' discrimination power in terms of the area under the receiver operating characteristic curve (AUC ROC). The AUC ROC is a widely adopted classification metric that quantifies the sensitivity and specificity of the model while is invariant to *a priori* class probabilities.

This paper is organized as follows. Section 2 outlines related scientific works, Section 3 provides an overview of the use case, and Section 4 details the methodology. Finally, section 5 presents and discusses the results obtained, while Section 6 concludes and describes future work.

2 RELATED WORK

The field of radiomics is concerned with extracting high-dimensional data from medical images, which can be mined to provide diagnoses and prognoses, assuming the image features reflect an underlying pathophysiology[16, 27, 28]. While the research on the field is experiencing exponential growth, multiple authors have warned about common issues affecting the quality and reproducibility of radiomics research and proposed several criteria that should be met to mitigate them (e.g., RQS, CLAIM, or TRIPOD)[10, 27, 32]. It has also been observed that the translation into clinical use has been slow[13].

Since the start of the COVID-19 pandemic, much research has been published highlighting how AI models could be used to issue COVID-19 diagnoses based on medical images. While much research was invested into transfer learning leveraging pre-trained deep learning models, or the use of deep learning models as feature extractors[24], some authors also experimented with handcrafted features[7]. Most common machine learning approaches involved the use of deep learning (end-to-end models, or pre-trained models for feature extraction)[14, 23, 34, 36, 43], Support Vector Machine (SVM)[4, 7, 14, 22, 23, 34, 36, 38, 43], k-Nearest Neighbors (kNN)[14, 22, 23, 38, 43], Random Forest (RF)[22, 23, 36], CART[22, 23, 36], Näive Bayes[22, 23], and Gradient Boosted Machines (GBM)[6, 22].

Two commonly faced challenges regarding COVID-19 diagnoses based on medical images are images scarcity and class imbalance. Given the heterogeneity of the datasets, it is not always possible to merge them[2, 7, 35]. Thus, some researchers successfully experimented using generative adversarial networks (GANs) to generate new images that comply with the existing patterns in the dataset[1, 34]. GANs provide means to learn deep representations from labeled data and generate new data samples based on a competition involving two models: a *generator*, learns to generate new images only from its interaction with the *discriminator*; and the *discriminator*, who has access to the real and synthetic data instances, and tries to tell the difference between them[3, 11]. While this method was first applied on images[17], new approaches were developed to adapt it for tabular data[41].

The fact that the classification categories are not approximately equally represented in a dataset can affect how the machine learning algorithms learn and their performance on unseen data, where the distribution can be different from the one observed in training data[8]. Due to these reasons, care must be taken to select metrics not sensitive to such imbalance. Among common strategies to deal with class imbalance, we find oversampling data methods, which aim to increase the number of data instances of the minority class to balance the dataset. Oversampling methods can add data instances from existing ones by replicating them (e.g., using a näive random sampler that draws new samples by randomly sampling with replacement from the available train samples), or by creating synthetic data instances (e.g., through SMOTE[9], ADASYN[19], or GANs). In addition to data oversampling, the *Focal Loss*[29] can be used on specific algorithms. The *Focal Loss* reshapes the cross-entropy loss to down-weight well-classified examples while focusing on the misclassified ones, achieving better discrimination. Finally, while the techniques mentioned above are useful for classification, we can reframe the problem as an anomaly detection problem, attempting to detect which data instances correspond to the minority class (anomaly).

Through the research we reviewed, we found a paper describing the use of SMOTE[14], and two papers using GANs[1, 34] for data augmentation at the image level. We found no paper performing a more extensive assessment of the class imbalance influence nor compared class imbalance strategies towards the COVID-19 detection models' outcomes. We propose utilizing data augmentation techniques, generating new embeddings instead of full images. Such an approach provides similar information in the embedding space as would be obtained from synthetic images while enabling widely used techniques for tabular data oversampling. Furthermore, in GANs, new data instances are cheaper to compute and store than would be if creating new images.

3 USE CASE

The research reported in this paper is done with images provided by the Medical Physics Research Group at the University of Ljubljana and made available as part of the RIS competition. The dataset was built from computed tomography (CT) scans obtained from three datasets reported in[18, 25, 33], that correspond to 289 healthy persons and 66 COVID-19 patients. Healthy persons are determined with a CT score between zero and five, while COVID-19 patients are considered those with a CT score equal to or higher than ten[15]. Each CT scan was segmented into twenty slices, resulting in 7.100 images with an axial view of the lungs, and annotated into two classes: COVID-19 and non-COVID-19. The visual inspection of CT scans aims to determine if the person was infected with the COVID-19 disease. Automating this task reduces manual work and speeds up the diagnosis.

4 METHODOLOGY

We propose using artificial intelligence for an automated COVID-19 diagnosis based on images obtained from CT scan segmentation, posing it as a binary classification problem. The discrimination capability of the models is measured with the AUC ROC metric with a cut threshold of 0.5.

We use the ResNet-18 model[20] for feature extraction, retrieving the vector produced by the Average Pooling layer. Since the vector consists of 512 features, we perform feature selection computing the features' mutual information and selecting the *top K* to avoid

[1] http://tiziano.fmf.uni-lj.si/

overfitting. To obtain K, we follow the equation $K = \sqrt{N}$ suggested by[21], where N is the number of data instances in the train set.

To evaluate the models' performance across different data augmentation strategies, we apply a stratified ten-fold cross-validation. Data augmentation is performed by introducing additional minority class data samples on the train folds. We consider five imbalance mitigation strategies: NONE (without data augmentation), RANDOM (näive random sampler), SMOTE, ADASYN, and CTGAN (GAN that enables the conditional generation of data instances based on a class label)[41]. No augmentation is performed on the test fold to ensure measurements are comparable. The performance of the data augmentation strategies is measured across eight machine learning algorithms: SVM, kNN, RF, CART, Gaussian Näive Bayes, Multilayer Perceptron (MLP), GBM, and Isolation Forest (IF)[30]. Finally, we compare the performance of the data augmentation scenarios computing the average AUC ROC across the test folds and assess if the difference is statistically significant by using the Wilcoxon signed-rank test, using a p-value of 0.05.

5 RESULTS AND ANALYSIS

When comparing the results across different imbalance mitigation strategies (see Table 1), we observed that data augmentation leads to inferior results in most cases. While this outcome was expected for IF (the minority class is no longer an outlier after data augmentation), we found that only the CART, MLP, and GBM algorithms achieved better performance with CTGAN data augmentation compared to the original dataset. Moreover, six algorithms achieved the best results when augmented with CTGAN compared to other data imbalance strategies (except NONE). We confirmed the AUC ROC differences between imbalanced datasets strategies were statistically significant, with a few exceptions: *SMOTE vs. ADASYN* for CART, MLP, and GBM; *NONE vs. RANDOM* for CART; *NONE vs. SMOTE* for Näive Bayes; *RANDOM vs. SMOTE* for SVM and RF; and *RANDOM and SMOTE vs. CTGAN* for SVM and IF. From the results obtained, we consider the CTGAN success can be attributed to the fact the generative model can learn over time to generate high-quality data instances based on the discriminator's feedback loop, while Näive random sampling reuses existing instances (providing little new information to the dataset), and the SMOTE and ADASYN algorithms generate new samples based on heuristics without learning capabilities.

We observed that GBM models trained with a Focal Loss achieved the best results in all datasets. Even when no data augmentation is performed and the RF achieves the best result, the difference is not statistically significant compared to the GBM model. The overall best performance was obtained with a GBM model trained over a dataset with CTGAN data augmentation. While the reasons behind the performance drop for the kNN, Näive Bayes, RF, and SVM models remain unclear, further investigation is required to clarify them. Nevertheless, we consider the CTGAN data augmentation on the embeddings space approach is promising.

6 CONCLUSION

This research presents a novel approach towards data augmentation in radiomics by generating new data instances in the embedding space rather than generating new images. We demonstrate that this approach leads to the best forecast outcomes with a GBM model trained with a Focal Loss on a dataset enriched with new CTGAN generated instances. Moreover, we compare this approach to other imbalanced data strategies, finding that Näive random oversampling, SMOTE, and ADASYN degrade the resulting models' performance compared to the original dataset. Future work will focus on further understanding the cases where the CTGAN data augmentation leads to poor results and provide an integral explainability model for machine learning classifiers that consume image embeddings.

ACKNOWLEDGMENTS

This work was supported by the Slovenian Research Agency. The authors acknowledge the *Medical Physics Research Group* at the University of Ljubljana[2] for providing the image segmentation data as part of the RIS competition[3].

REFERENCES

[1] Erdi Acar, Engin Şahin, and İhsan Yılmaz. 2021. Improving effectiveness of different deep learning-based models for detecting COVID-19 from computed tomography (CT) images. *Neural Computing and Applications* (2021), 1–21.
[2] Parnian Afshar, Shahin Heidarian, Nastaran Enshaei, Farnoosh Naderkhani, Moezedin Javad Rafiee, Anastasia Oikonomou, Faranak Babaki Fard, Kaveh Samimi, Konstantinos N Plataniotis, and Arash Mohammadi. 2021. COVID-CT-MD, COVID-19 computed tomography scan dataset applicable in machine learning and deep learning. *Scientific Data* 8, 1 (2021), 1–8.
[3] Alankrita Aggarwal, Mamta Mittal, and Gopi Battineni. 2021. Generative adversarial network: An overview of theory and applications. *International Journal of Information Management Data Insights* (2021), 100004.
[4] Dhurgham Al-Karawi, Shakir Al-Zaidi, Nisreen Polus, and Sabah Jassim. 2020. Machine learning analysis of chest CT scan images as a complementary digital test of coronavirus (COVID-19) patients. *MedRxiv* (2020).
[5] William E Allen, Han Altae-Tran, James Briggs, Xin Jin, Glen McGee, Andy Shi, Rumya Raghavan, Mireille Kamariza, Nicole Nova, Albert Pereta, et al. 2020. Population-scale longitudinal mapping of COVID-19 symptoms, behaviour and testing. *Nature human behaviour* 4, 9 (2020), 972–982.
[6] Eduardo J Mortani Barbosa, Bogdan Georgescu, Shikha Chaganti, Gorka Bastarrika Aleman, Jordi Broncano Cabrero, Guillaume Chabin, Thomas Flohr, Philippe Grenier, Sasa Grbic, Nakul Gupta, et al. 2021. Machine learning automatically detects COVID-19 using chest CTs in a large multicenter cohort. *European radiology* (2021), 1–11.
[7] Mucahid Barstugan, Umut Ozkaya, and Saban Ozturk. 2020. Coronavirus (covid-19) classification using ct images by machine learning methods. *arXiv preprint arXiv:2003.09424* (2020).
[8] Nitesh V Chawla. 2009. Data mining for imbalanced datasets: An overview. *Data mining and knowledge discovery handbook* (2009), 875–886.
[9] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research* 16 (2002), 321–357.
[10] Gary S Collins, Johannes B Reitsma, Douglas G Altman, and Karel GM Moons. 2015. Transparent reporting of a multivariable prediction model for individual prognosis or diagnosis (TRIPOD): the TRIPOD statement. *Journal of British Surgery* 102, 3 (2015), 148–158.
[11] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. 2018. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine* 35, 1 (2018), 53–65.
[12] Thays Maria Costa de Lucena, Ariane Fernandes da Silva Santos, Brenda Regina de Lima, Maria Eduarda de Albuquerque Borborema, and Jaqueline de Azevêdo Silva. 2020. Mechanism of inflammatory response in associated comorbidities in COVID-19. *Diabetes & Metabolic Syndrome: Clinical Research & Reviews* 14, 4 (2020), 597–600.
[13] Daniel Pinto Dos Santos, Matthias Dietzel, and Bettina Baessler. 2021. A decade of radiomics research: are images really data or just patterns in the noise?
[14] El-Sayed M El-Kenawy, Abdelhameed Ibrahim, Seyedali Mirjalili, Marwa Metwally Eid, and Sherif E Hussein. 2020. Novel feature selection and voting classifier algorithms for COVID-19 classification in CT images. *IEEE Access* 8 (2020), 179317–179335.

[2] https://medfiz.si/en
[3] http://tiziano.fmf.uni-lj.si/

Class Imbalance Mitigation Strategies	CART	IF	kNN	MLP	Naive Bayes	RF	SVM	GBM
NONE	0,6429	0,6802	0,8504	0,7879	0,6653	**0,8601**	0,8066	*0,8555*
RANDOM	0,6402	0,5215	0,7846	*0,7993*	0,6464	0,6691	0,6888	**0,8150**
SMOTE	0,6147	0,5607	0,6813	*0,7663*	0,6590	0,6660	0,6817	**0,7826**
ADASYN	0,6020	0,5863	0,6660	*0,7655*	0,6282	0,6435	0,6652	**0,7787**
CTGAN	0,7401	0,5340	0,8118	*0,8419*	0,6395	0,7090	0,6896	**0,8871**

Table 1: Average AUC ROC values obtained across the ten cross-validation folds. Best results are bolded, second-best results are highlighted in italics.

[15] Marco Francone, Franco Iafrate, Giorgio Maria Masci, Simona Coco, Francesco Cilia, Lucia Manganaro, Valeria Panebianco, Chiara Andreoli, Maria Chiara Colaiacomo, Maria Antonella Zingaropoli, et al. 2020. Chest CT score in COVID-19 patients: correlation with disease severity and short-term prognosis. *European radiology* 30, 12 (2020), 6808–6817.

[16] Robert J Gillies, Paul E Kinahan, and Hedvig Hricak. 2016. Radiomics: images are more than pictures, they are data. *Radiology* 278, 2 (2016), 563–577.

[17] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Advances in neural information processing systems* 27 (2014).

[18] Stephanie A Harmon, Thomas H Sanford, Sheng Xu, Evrim B Turkbey, Holger Roth, Ziyue Xu, Dong Yang, Andriy Myronenko, Victoria Anderson, Amel Amalou, et al. 2020. Artificial intelligence for the detection of COVID-19 pneumonia on chest CT using multinational datasets. *Nature communications* 11, 1 (2020), 1–7.

[19] Haibo He, Yang Bai, Edwardo A Garcia, and Shutao Li. 2008. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*. IEEE, 1322–1328.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[21] Jianping Hua, Zixiang Xiong, James Lowey, Edward Suh, and Edward R Dougherty. 2005. Optimal number of features as a function of sample size for various classification rules. *Bioinformatics* 21, 8 (2005), 1509–1515.

[22] Lal Hussain, Tony Nguyen, Haifang Li, Adeel A Abbasi, Kashif J Lone, Zirun Zhao, Mahnoor Zaib, Anne Chen, and Tim Q Duong. 2020. Machine-learning classification of texture features of portable chest X-ray accurately classifies COVID-19 lung infection. *BioMedical Engineering OnLine* 19, 1 (2020), 1–18.

[23] Seifedine Kadry, Venkatesan Rajinikanth, Seungmin Rho, Nadaradjane Sri Madhava Raja, Vaddi Seshagiri Rao, and Krishnan Palani Thanaraj. 2020. Development of a machine-learning system to classify lung ct scan images into normal/covid-19 class. *arXiv preprint arXiv:2004.13122* (2020).

[24] Sara Hosseinzadeh Kassania, Peyman Hosseinzadeh Kassanib, Michal J Wesolowskic, Kevin A Schneidera, and Ralph Detersa. 2021. Automatic detection of coronavirus disease (COVID-19) in X-ray and CT images: a machine learning based approach. *Biocybernetics and Biomedical Engineering* 41, 3 (2021), 867–879.

[25] Michael T Kassin, Nicole Varble, Maxime Blain, Sheng Xu, Evrim B Turkbey, Stephanie Harmon, Dong Yang, Ziyue Xu, Holger Roth, Daguang Xu, et al. 2021. Generalized chest CT and lab curves throughout the course of COVID-19. *Scientific reports* 11, 1 (2021), 1–13.

[26] Virendra Kumar, Yuhua Gu, Satrajit Basu, Anders Berglund, Steven A Eschrich, Matthew B Schabath, Kenneth Forster, Hugo JWL Aerts, Andre Dekker, David Fenstermacher, et al. 2012. Radiomics: the process and the challenges. *Magnetic resonance imaging* 30, 9 (2012), 1234–1248.

[27] Philippe Lambin, Ralph TH Leijenaar, Timo M Deist, Jurgen Peerlings, Evelyn EC De Jong, Janita Van Timmeren, Sebastian Sanduleanu, Ruben THM Larue, Aniek JG Even, Arthur Jochems, et al. 2017. Radiomics: the bridge between medical imaging and personalized medicine. *Nature reviews Clinical oncology* 14, 12 (2017), 749–762.

[28] Philippe Lambin, Emmanuel Rios-Velazquez, Ralph Leijenaar, Sara Carvalho, Ruud GPM Van Stiphout, Patrick Granton, Catharina ML Zegers, Robert Gillies, Ronald Boellard, André Dekker, et al. 2012. Radiomics: extracting more information from medical images using advanced feature analysis. *European journal of cancer* 48, 4 (2012), 441–446.

[29] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*. 2980–2988.

[30] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. 2008. Isolation forest. In *2008 eighth ieee international conference on data mining*. IEEE, 413–422.

[31] Hossein Mohammad-Rahimi, Mohadeseh Nadimi, Azadeh Ghalyanchi-Langeroudi, Mohammad Taheri, and Soudeh Ghafouri-Fard. 2021. Application of machine learning in diagnosis of COVID-19 through X-ray and CT images: a scoping review. *Frontiers in cardiovascular medicine* 8 (2021), 185.

[32] John Mongan, Linda Moy, and Charles E Kahn Jr. 2020. Checklist for artificial intelligence in medical imaging (CLAIM): a guide for authors and reviewers.

[33] Sergey P Morozov, Anna E Andreychenko, Ivan A Blokhin, Pavel B Gelezhe, Anna P Gonchar, Alexander E Nikolaev, Nikolay A Pavlov, Valeria Yu Chernina, and Victor A Gombolevskiy. 2020. Mosmeddata: data set of 1110 chest ct scans performed during the covid-19 epidemic. *Digital Diagnostics* 1, 1 (2020), 49–59.

[34] Jawad Rasheed, Alaa Ali Hameed, Chawki Djeddi, Akhtar Jamil, and Fadi Al-Turjman. 2021. A machine learning-based framework for diagnosis of COVID-19 from chest X-ray images. *Interdisciplinary Sciences: Computational Life Sciences* 13, 1 (2021), 103–117.

[35] Michael Roberts, Derek Driggs, Matthew Thorpe, Julian Gilbey, Michael Yeung, Stephan Ursprung, Angelica I Aviles-Rivero, Christian Etmann, Cathal McCague, Lucian Beer, et al. 2021. Common pitfalls and recommendations for using machine learning to detect and prognosticate for COVID-19 using chest radiographs and CT scans. *Nature Machine Intelligence* 3, 3 (2021), 199–217.

[36] Prottoy Saha, Muhammad Sheikh Sadi, and Md Milon Islam. 2021. EMCNet: Automated COVID-19 diagnosis from X-ray images using convolutional neural network and ensemble of machine learning classifiers. *Informatics in medicine unlocked* 22 (2021), 100505.

[37] Adekunle Sanyaolu, Chuku Okorie, Aleksandra Marinkovic, Risha Patidar, Kokab Younis, Priyank Desai, Zaheeda Hosein, Inderbir Padda, Jasmine Mangat, and Mohsin Altaf. 2020. Comorbidity and its impact on patients with COVID-19. *SN comprehensive clinical medicine* (2020), 1–8.

[38] Ahmet Sayglı. 2021. A new approach for computer-aided detection of coronavirus (COVID-19) from CT and X-ray images using machine learning methods. *Applied Soft Computing* 105 (2021), 107323.

[39] H Swapnarekha, Himansu Sekhar Behera, Janmenjoy Nayak, and Bighnaraj Naik. 2020. Role of intelligent computing in COVID-19 prognosis: A state-of-the-art review. *Chaos, Solitons & Fractals* 138 (2020), 109947.

[40] Tianbing Wang, Zhe Du, Fengxue Zhu, Zhaolong Cao, Youzhong An, Yan Gao, and Baoguo Jiang. 2020. Comorbidities and multi-organ injuries in the treatment of COVID-19. *The Lancet* 395, 10228 (2020), e52.

[41] Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. 2019. Modeling tabular data using conditional gan. *arXiv preprint arXiv:1907.00503* (2019).

[42] Jing Yang, Ya Zheng, Xi Gou, Ke Pu, Zhaofeng Chen, Qinghong Guo, Rui Ji, Haojia Wang, Yuping Wang, and Yongning Zhou. 2020. Prevalence of comorbidities in the novel Wuhan coronavirus (COVID-19) infection: a systematic review and meta-analysis. *Int J Infect Dis* 10, 10.1016 (2020).

[43] Huseyin Yasar and Murat Ceylan. 2021. A novel comparative study for detection of Covid-19 on CT lung images using texture analysis, machine learning, and deep learning methods. *Multimedia Tools and Applications* 80, 4 (2021), 5423–5447.

[44] Stephen SF Yip and Hugo JWL Aerts. 2016. Applications and limitations of radiomics. *Physics in Medicine & Biology* 61, 13 (2016), R150.

[45] Kun-Hsing Yu, Andrew L Beam, and Isaac S Kohane. 2018. Artificial intelligence in healthcare. *Nature biomedical engineering* 2, 10 (2018), 719–731.

Observing Water-Related Events for Evidence-Based Decision-Making

Joao Pita Costa * *****, M. Besher Massri *, Inna Novalija *, Ignacio Casals del Busto **, Iulian Mocanu ***, Maurizio Rossi ****, Jan Šturm *, Eva Erzin *, Alenka Guček *, Matej Posinković *, Marko Grobelnik * *****

* Institute Jozef Stefan, Slovenia, ** Aguas del Alicante, Spain, *** Apa Braila, Romania, **** Ville de Carouge, Switzerland, ***** Quintelligence, Slovenia

ABSTRACT

With the awareness of a changing climate impacting our sustainability, and in line with the European Green Deal initiative or the Sustainable Development Goal 6 addressing water, the industry, society and local governments are requiring reliable and comprehensive technology that can provide them an overview to water events to anticipate problems and the tools to analyse best practices appropriate to solve them. This paper presents the NAIADES Water Observatory (NOW), a digital solution offering a series of analysis and visualisations of water-related topics, helping users to extract important insights in relation to the water sector. Taking advantage of heterogeneous data sources, from the media and social media landscape, to published research and global/local indicators. Through collaboration with local water resource management institutions, the NWO was configured to local priorities and ingests local datasets to better fit the needs of decision-makers.

CCS CONCEPTS

• Real-time systems • Data management systems • Life and medical science

KEYWORDS

Water Resource Management, Smart Water, Observatory, Water Digital Twin, Elasticsearch, Streamstory

1 Introduction

The water sector is facing rapid development towards the smart digitalisation of resources, much motivated and supported by the UN's global initiative for the Sustainable Development Goal 6. In that context, the efforts to address the specific challenges related to water management data and priorities multiply globally. There are several "digital twin" systems dedicated to water, each of which focuses on the different aspects of the digitalisation of signals to support water management companies, as well as water "observatories". These are usually meant as Geographical Information Systems that showcase the different aspects of water resources through time.

Within the scope of the European Commission-funded project NAIADES [1] focusing on the automation of the water resource management and environmental monitoring, we propose a slightly different approach that integrates heterogeneous data sources to try and solve common research questions, as well as to support water management companies in their current problems. This solution is named NAIADES Water Observatory (NWO), available at naiades.ijs.si, putting together: (i) real-time information from multilingual world news on water topics; (ii) data visualisation of water-related indicators through time, sourced from the datasets associated with the Sustainable Development Goal 6 (water) and other UN data (see Figure 1); and (iii) scientific knowledge from published biomedical research on water-related topics (e.g., water contamination).

Due to the rapidly growing awareness of the sustainability challenges that we are facing in Europe and worldwide in the context of water resource management, there has been much work done to develop systems that are able to collect information about the available water and even simulate and forecast that in the near future. But these are usually geolocation-based systems ingesting water-related data to enable real-time monitoring of resources and usage [x] [y] [z], and thus much different than the water observatory that we are proposing in this paper. The typical example is GoAigua system [4], a digital twin technology allowing, e.g., the city of Valencia to optimize its water management at the network level, improving efficiency in daily operations, plan real-time scenarios, and make some prediction on its future behaviour [5].

Figure 1: **Visualisation of water-related indicators within Spain to complement the global indicators view ingesting data from, e.g., U.N. and the World Bank.**

2 A data-driven solution for water events

The proposed Water Observatory enables extraction of insightful water-related information, configured to use case priorities and needs from the data integration of

heterogeneous sources. This includes information from social media when the weather is favourable for floods and the historical information from news and published research on these weather-related events and how to make better decisions to solve them.

This is complemented by data ingested from global and local indicators (i.e., datasets at regional level), showcasing the observation of water-related datasets linked to SDG 6 at global and country levels that can help us observe changes and trends. The NAIADES Water Observatory enables the user to explore the information provided by published science and the success stories that can be used in decision-making and water education at the local level (i.e., showcasing the resources and problematics of the region).

In this approach, the water data sensing is done over dynamic open data sources that serve as digital sensors (news, social media, indicators, publications, weather forecasts). This data is then integrated and visualised, each in its tab, addressing specific topics of interest. The observatory is thus composed of all that heterogeneous data coming in at different frequencies. The interactions between those data sources to solve common problems make it a Water Digital Twin. The envisioned examples include the analysis of best practices in water events in, e.g. Braila, identified in the news and explored over the published research, or the alerts triggered by weather conditions and observed over social media on a water event. The questions we are trying to solve with this innovative technology are, e.g., if we can predict water shortages in a certain region given the historical data; or if we can identify early signals of water-related problems from social media (see Figure 2).

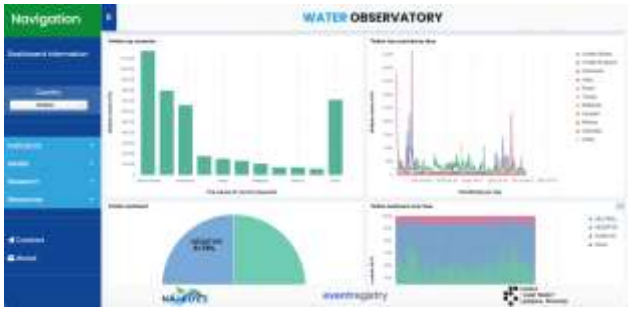

Figure 2: **Analysis of the sentiment in water-related posts in Twitter and the relation to consumer satisfaction and water-related events**

All of the views of this observatory, each of which represents digital solutions on their own, are configured to the local priorities of the NAIADES users as a Proof of Concept, showing that each can address specific conditions.

- Indicators: adding to the global UN indicators, we are ingesting curated open datasets that have regional information about water topics of interest to the stakeholder
- Media: each location has its own news and social media streams configured to priorities and aspects of the news that stakeholders define as topics of interest (e.g. floods)
- Research: similarly to the media sources, the research topics allow for some customisation to fit the needs of the local user better
- Resources: the natural resources information provided for exploration is geolocated to the regions of interest to the user of the platform

It is relatively easy to include new use cases and corresponding workspaces after the discussions on user priorities that will allow us to configure the information presented and making it meaningful.

3 Addressing the challenges of tomorrow

With the range of views provided at the observatory, the problems addressed can be of complex nature and cover a range of concerns and workflows. The different ICT capabilities available across the water sector require intuitive and meaningful technologies to ensure the usefulness of the contribution to the Community. The target users of the NWO seem to belong to three main scenarios with different workflows that can be supported by the developed technology:

1. Water resource management: using the provided information in the resolution of problems related to weather events to understand how their actions are perceived by the consumers and to explore successful scenarios in similar cases
2. Local governments: to help evidence-based decision-making using open data, better synchronise to SDG6 and other guidelines and evaluate commitments in time
3. General public: for water education with a local context, in aspects that matter to the local population, based on parts of the Water Observatory that can be open to public

The priorities in the European Union are rapidly changing towards sustainability and environmental efficiency, transversally to most domains of action. The European Commission's Green Deal [3] aiming for a climate-neutral Europe by 2050 and boosting the economy through green technology provides a new framework to understand and position water resource management in the context of the challenges of tomorrow. The NAIADES Water Observatory will not only contribute to the improvement of European sustainability in water-related matters but will also assign the local actors on the water resource management an active role in that. The NAIADES Water Observatory provides the user of the NAIADES platform, as earlier extensively discussed, with the global and local insight that can be transformed into business intelligence, and help companies to steer their strategies towards customer satisfaction. We will be

describing selected views of this observatory through the verticals (or views) News – Indicators – Biomedical, first at the level of the specific dashboards that constitute the tabs in the online instance, and then by the extended exploratory instances, including public instances and APIs, for each of the three verticals.

Figure 3: **The global view of the pilot 1 over usage and data sources.**

These dashboards come together to provide the user with a global perspective in real-time, where five different tiers of usability are made available (see Figure 3). The tiers allow for the extended usability of the Water Observatory, Transversally to the data sources available.

4 System description and architecture

The NWO offers user exploratory dashboards for the further investigation over news, to get deeper into the indicators ingested, and to explore the biomedical research on water contamination in detail. Moreover, each of the three dashboards have versions built to be exposed by, e.g., iframe through a publicly available channel that can be used for integration in high management KPI-monitoring dashboards. Furthermore, we also offer a part of the information in these through APIs easily integrable with our own systems.

The *Indicators* view provides the user with interactive data exploration tools that allow for the KPI-monitoring over several water-related topics that include the SDG 6, the World Bank Open Data, the UN data, etc. In this module we also ingest regional data sources that include local indicators, addressing the user's priorities. Considering their well-established data types, the data integration is possible and, whenever limitations appear due to lack or poor quality of the data, the dataset is pre-processed to allow for data completion (whenever possible), or at least the improvement of data quality.

The *Media* view provides the user with the real-time news monitoring over water-related topics (such as Water Scarcity and Water Contamination), and the analysis of water-related tweets based on data visualisation modules. Based on the news engine *Eventregistry* [7] this view provides the system with a continuous stream of news articles, sourced from RSS-enabled sites across the world. From the data management module the real-time news data is accessed by the news dashboard that can be configured by the NAIADES user to tune the topics of interest in the configuration web app. To further explore a water-related topic, the NWO provides a dashboard for the analysis of social media posts in Twitter (see Figure 2), collected in a real-time frequency, where sentiment is analysed, related concepts are extracted and it is possible to access the raw tweets or apply several filters.

Finally, the biomedical module allows for the exhaustive exploration of water contamination information from scientific research articles published worldwide and available through the MEDLINE biomedical open dataset [9] and the Microsoft Academic Graph [8]. The MEDLINE dataset is collected from the official FTP source made available by the North American National Library of Medicine (NLM) over an XML dump and uploaded to the elasticSearch data management system through a python script, the Microsoft Academic Graph dataset is collected from an Azure container with the data biweekly updated by the Microsoft Research team. The data management is based on the elasticsearch technology [2, useful for both the interactive data visualisations and the Indicators Explorer view. The latter allows the NAIADES user to explore the raw data through template visualisations, use a Lucene-based query that can leverage the loaded metadata, and easily build visualisation modules that can define a new dashboard of data visualisation modules. The dataset is then called over and HTTP API by the SearchPoint technology [6] to load the dataset and respective metadata. thus allowing for powerful Lucene-based queries and further interaction over a movable pointer. This will lead to the refinement of the search of information that can then be extended over the Biomedical Explorer, which feeds over the same dataset through Kibana, but also allows for the analysis of raw data, or the easy construction of data visualisation modules from templates, and for an interactive data visualisation dashboard. All the mentioned dashboards can be made publicly available through, e.g., iframe to be integrated in high-management KPI monitors.

Figure 4: **System architecture of the NAIADES Water Observatory showcasing the relation between used technologies and NOW views**

5. Conclusions and further work

In this paper we discussed the technological development and research opportunities motivated by the emerging need to support decision-makers with evidence from open data that can retract best practices and answer questions from the collected data, bringing the digitalisation of the water sector to a new level.

The potential to ingest complementary local data and configure global sources to parameters addressing local priorities provides a local dimension that is being explored close to the priorities of the NAIADES data providers within water resource management institutions. It will also be exploring the insights driven by the appropriate aspects of chosen datasets, e.g., between news data and focused interactions through Twitter for weather-related events when the weather is likely to be favourable to their cause (see Figure 5). There are many systems that can collect business intelligence data, but we believe that the "digital twin"-type of insight is in the interaction between these data streams.

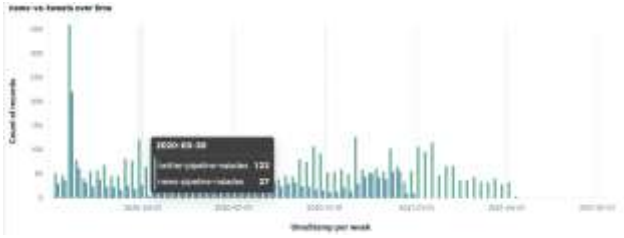

Figure 5: **Preliminary data analysis of the relation between news and tweets on water-related events and their relations with other topics (e.g., weather).**

Further development to the NAIADES Water Observatory, will be providing the users with tools to explore the impact of natural resources as, e.g., the weather, as well as predictions on the levels of the available bodies of water, based on ingested weather data from the ECMWF (on humidity, temperature and rainfall) and other open data sources. This will help the users to have some insight on the impact of the climate crisis in regions that directly relate to their water resources. We will use a sophisticated engine - Streamstory [6][10] - to explore the states of that weather-related data and short/medium term predictions on aspects of that data (see Figure 6).

Figure 6: **The multi time-series analysis of the weather parameters, using Markov chains in complex data visualisation through the Streamstory technology** [9]**.**

ACKNOWLEDGMENTS

We thank the support of the European Commission on the H2020 NAIADES project (GA nr. 820985).

REFERENCES

[1] CORDIS, "NAIADES Project". [Online]. Available: https://cordis.europa.eu/project/id/820985 [Accessed 1 9 2020].
[2] Elasticsearch, "Elasticsearch," 2020. [Online]. Available: https://www.elastic.co/elasticsearch/. [Accessed 1 9 2020].
[3] European Commission, "European Green Deal," 2019. [Online]. Available: https://ec.europa.eu/info/strategy/priorities-2019-2024/european-green-deal_en. [Accessed 1 9 2020].
[4] Idrica, "GoAigua: Smart Water for a Better World," 2020. [Online]. Available: https://www.idrica.com/goaigua/. [Accessed 1 9 2020].
[5] Idrica, "Digital Twin: implementation and benefits for the water sector," 19 2 2020. [Online]. Available: https://www.idrica.com/blog/digital-twin-implementation-benefits-water-sector/. [Accessed 1 9 2020].
[6] Institute Jozef Stefan, "Streamstory". [Online]. Available: http://streamstory.ijs.si/. [Accessed 26 8 2021]
[7] G. Leban, B. Fortuna, J. Brank and M. Grobelnik, "Event registry: learning about world events from news," Proceedings of the 23rd International Conference on World Wide Web, pp. 107-110, 2014.
[8] Microsoft, "Microsoft Academic Graph". [Online]. Available: https://www.microsoft.com/en-us/research/project/microsoft-academic-graph/. [Accessed 26 8 2021]
[9] National Library of Medicine, "MEDLINE". [Online]. Available: https://www.nlm.nih.gov/medline/medline_overview.html. [Accessed 26 8 2021]
[10] L. Stopar, P. Škraba, M. Grobelnik, and D. Mladenić (2018). StreamStory: Exploring Multivariate Time Series on Multiple Scales. IEEE transactions on visualization and computer graphics 25.4: 1788-1802.

Anomaly Detection on Live Water Pressure Data Stream

Gal Petkovšek
Jožef Stefan Institute
Jamova 39, 1000 Ljubljana, Slovenia
gal.petkovsek@ijs.si

Matic Erznožnik
Jožef Stefan Institute
Jamova 39, 1000 Ljubljana, Slovenia
matic.erznoznik@ijs.si

Klemen Kenda
Jožef Stefan Institute
Jožef Stefan International Postgraduate School
Jamova 39, 1000 Ljubljana, Slovenia
klemen.kenda@ijs.si

ABSTRACT
We present the application of several anomaly detection algorithms to water pressure data streams. We evaluate their quality on unlabelled data sets using agreement rates. The applied algorithms are the Generative Adversarial Network (GAN), DBSCAN, Welford's algorithm and Facebook Prophet. We found that GAN performed best.

Keywords
water management, machine learning, anomaly detection

1. INTRODUCTION
In last decades, Internet of Things (IoT) has penetrated and shaped several fields such as energy management, traffic, health care and others. The water sector is, however, still implementing IoT solutions that will improve the water management with features such as real-time consumption prediction, leakage detection, water quality estimation and others.

In the presented work, we focus on the anomaly detection on the live water pressure data stream from the town of Braila (Romania). The overall goal of the research is to detect leakage points in the city's water distribution network. To detect the presence of a leakage in the system we apply an anomaly detection algorithm to the water pressure data stream. We considered several such algorithms, which were applied and evaluated on four data streams obtained from four pressure sensors. Our goal was to find the algorithm which returns the best results. Since the data is not labeled (regular or anomalous), the estimation of accuracy was done with a method considering relative agreement among selected algorithms [1]. The anomaly detection algorithms that were tested were GAN (generative adversarial networks) [6], DBSCAN [10], Welford's algorithm [9] and anomaly detection with Facebook Prophet [11]. It is important to note that first three algorithms consider the data stream as an actual live stream. This means that they consume one sample at a time (or a feature vector containing multiple past values, enrichment values and contextual data) and declare it regular or anomalous as the algorithms were intended to do in production. In contrast, the Facebook Prophet consumes the whole data stream as a batch and labels all the samples together. This makes it unusable in production (in this setting), however it is included in the experiment since it can help to estimate the accuracy of other algorithms.

Anomaly detection on time series is a well researched field. The algorithms in this paper were already considered in the related work in different settings and for different time series.

Anomaly detection can be used by estimating the expected regular interval in the upcoming measurement. This can be achieved in an incremental fashion with a simple short-term prediction model, for example with Kalman filter [7], or with a more advanced approach, based on time-series modeling [11]. The latter can be used in several settings, for example in detecting air temperature anomalies in the sewer systems [12].

DBSCAN [10] is a data clustering algorithm that can be applied in frequently changing data sets. Its incremental version [5] can be used in a streaming setting. The potential of the algorithm for anomaly detection has been demonstrated in several use cases, for example in detecting air temperature anomalies [3].

The paper that demonstrated the use of Generative Adversarial Networks for anomaly detection on data stream is fairly recent [6]. The authors have shown that this approach can outperform several other baselines on data sets obtained from NASA, Yahoo, Amazon etc. They introduced different measures of evaluating the reconstruction accuracy, which we tried to improve upon in our paper.

In this work, we use the already established anomaly detection approaches and compare their performance on an unlabeled water pressure data stream from a water distribution network. A more detailed description of the algorithms is given in the Methodology section. We argue that the relative agreement approach [1] improves the anomaly detection performance, which we demonstrate by manual evaluation of the results.

2. DATA AND DATA PREPROCESSING
We demonstrate our anomaly detection methodology on four data sets. Each of the data sets represents the pressure values of one of the sensors, which are located at different points in Braila's water distribution network. The sensors are labeled as '5770', '5771', '5772' and '5773'. The data sets contain between 10 and 11 thousand instances, which are spaced in 15 minute intervals, so about 100 days-worth of data. The data was first pre-processed to remove any duplicated points and 'holes' in the data which were formed as a consequence of sensor down-time. When working with data streams, this process should be done automatically to

avoid any incorrect analysis when feeding the data into the anomaly detection algorithms. Each of the four data sets was split into a training and evaluation part. The training sets consisted of the first 2000 data points and the evaluation sets contained all the rest. This is done so that the algorithms which require training can be trained on one part of the data and evaluated on the other (GAN, DBSCAN).

3. METHODOLOGY

3.1 Evaluation of algorithms

Evaluation of the performance of algorithms on unlabelled data always represents a challenge. Since we are working with such data an actual calculation of accuracy scores would require manual labelling of the data instances. To avoid this time-consuming process, we use a method for estimating error rates (ratio of wrong classifications to the total number of instances) from the agreement rates of multiple algorithms. Agreement rate of two classifiers f_i and f_j is defined in the following way:

$$a_{\{i,j\}} = \frac{1}{S}\sum_{s=1}^{S}\mathbb{I}\{f_i(X_s) = f_j(X_s)\}$$

where $X_1, ..., X_S$ are unlabeled samples. The calculated agreement rates are then inserted into the following equations:

$$a\{i,j\} = 1 - e_{\{i\}} - e_{\{j\}} + 2e_{\{i,j\}}$$

Here we assume that the functions make independent errors we can substitute $e\{i,j\}$ with $e_{\{i\}}e_{\{j\}}$. With such a system of equations we can then calculate error rates using some root-finding algorithm. Such an approach has been previously used for the evaluation of classifiers on an unlabelled dataset [1]. Therefore we consider the anomaly detection algorithm as a binary classifier and use the aforementioned method for the comparison of different algorithms. Additionally, two important assumptions were made. Firstly, we assumed that the anomaly detection algorithms were independent and secondly, that each of those algorithms performs better than a random classifier.
Since the estimated performance of one algorithm depends on the output of the others it was important that the algorithms yield a similar percentage of anomalies. In other words, the algorithms are tuned to have similar predicted positive condition rate ($PPCR = \frac{FP+TP}{FP+TP+FN+TN}$). For most data streams this means that 1%-3% of the samples are labelled as anomalous.

3.2 GAN

The Generative Adversarial Network (GAN)[6] is an unsupervised machine learning approach to anomaly detection. An encoder-decoder structure of the neural network is used to first encode the input data point and then decode the encoded one. The model learns to reconstruct the input data point as closely as possible. The idea is that the reconstruction should be better if the input data is 'normal' and worse if it is abnormal/anomalous. We use an input vector, which is composed of 10 consecutive values of the uni-variate data stream. We then compare the input vector to the reconstructed one using the mean squared error (MSE) metric. We classify the data point as 'normal' if the value of the MSE is below the defined threshold. [6] calculated the thresholds using sliding windows on reconstruction errors (4 standard deviations from the mean of the window). We used a slightly different approach using the moving average multiplied by a constant as the threshold. This proved to be easier to implement on our live data stream use-case.

3.3 DBSCAN

DBSCAN [4] is a well-known data clustering algorithm. It groups together points, which are close together based on Euclidean distance. The group with the largest number of points in our case are considered 'normal', and the lower-density groups are outliers which are then labeled as an anomaly. The ϵ parameter which measures how close the points should be for them to still be considered of the same group, can be adjusted based on the data set, and the desired sensitivity of the algorithm. For DBSCAN we also use an input vector composed of consecutive pressure values. In this case, we discovered that a vector of 5-6 values works best.

3.4 Welford's algorithm

Welford's algorithm gets its name from the Welford's method for online estimation of mean and variance. A very simple anomaly detection approach [9] can then be constructed by defining the upper and lower limits (UL and LL) of "normal" data as a function of mean and variance:

$$UL = mean + X * variance$$

$$LL = mean - X * variance$$

X is fixed and determines the threshold band. Any instance which falls out of that band is labeled as an anomaly. Instances can then be input into the algorithm one by one to be labeled and after each the mean and the variance (consequently UL and LL also) are updated.
For this experiment the actual Welford's method was not used since the mean and variance were computed from the last 1500 samples so that they would better adapt to the new samples. Note that the first 1500 samples therefore could not be labeled; however, this was not a problem since most of the other approaches required 2000 samples for fitting the models and the evaluation was therefore done on the remaining stream. However, the upper and lower limits of the interval were still computed as shown above with the value of $X = 2.2$.

3.5 Facebook Prophet

Facebook Prophet is an algorithm for time series forecasting that works especially well on data streams with multiple seasonalities [8]. Prophet also works well with missing data which makes it a good candidate for the problem at hand. After fitting the model it can make predictions for a chosen set of timestamps presented to it. Furthermore besides the prediction it also outputs upper and lower limits of the confidence interval for every sample. Ashrapov [2] demonstrates the implementation of an anomaly detection algorithm which uses this property to classify the samples inside the confidence interval as regular and the rest as anomalies. The model is fitted on the entire data set and then makes predictions on the same data set, providing both the anomaly detection and the confidence interval.

4. RESULTS

The results of the algorithms for data stream from sensor 5770 are presented in Figures 1, 2, 3 and 4. The charts show the raw values obtained from the pressure sensors, indicating the points which are labeled as anomalies with red points. Since the data sets are unlabelled it is hard to assess the accuracy of each algorithm based on anomaly visualizations alone, but we do notice some similarities and some differences. All of the algorithms are good at identifying obvious outliers (points which fall far out of the 'normal' range). The difference between the algorithms can be noticed when classifying points closer to the normal range. For example Welford's algorithm tends to label points as anomalies at the peaks of daily pressure fluctuation, which might not be ideal since we know that this behaviour can be considered normal. More sophisticated algorithms such as GAN and Prophet were also able to identify more "subtle" anomalies.

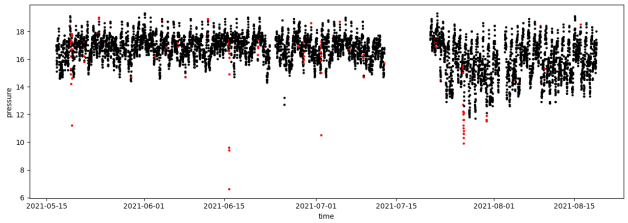

Figure 1: Anomalies found using GAN on data stream from sensor 5770.

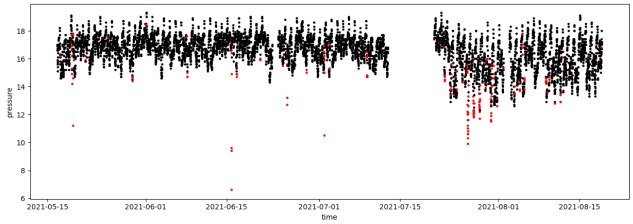

Figure 2: Anomalies found using DBSCAN on datastream from sensor 5770

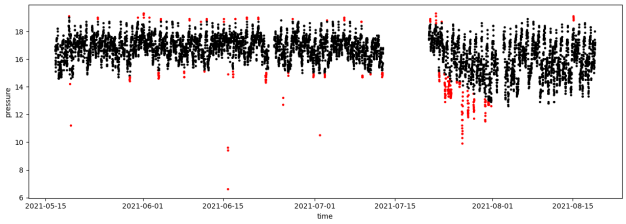

Figure 3: Anomalies found using Welford's algorithm on datastream from sensor 5770.

The recall of each algorithm can be increased or decreased by modifying parameters and thresholds. Since the data sets are unlabeled, it is hard to determine the optimal parameters. We decided to tune the algorithms to have similar recall of 1 - 3%, as we deemed that this would make the comparison of the algorithms the most fair. In Table 1 the shares of anomalies are presented for each separate data stream.

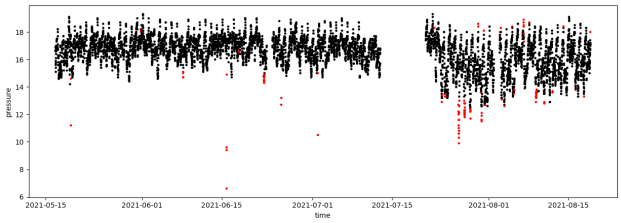

Figure 4: Anomalies found using Facebook Prophet on datastream from sensor 5770.

Algorithm	5770 anomaly share	5771 anomaly share	5772 anomaly share	5773 anomaly share
GAN	1.42%	0.99%	0.77%	1.13%
DBSCAN	2.63%	2.82%	2.73%	2.85%
Welford's algorithm	3.39%	3.41%	1.66%	3.16%
Facebook Prophet	1.66%	1.13%	0.46%	1.40%

Table 1: Shares of anomalies for all four data streams.

The error rates calculated from agreement rates are shown in Table 1 for each of the data streams. Since we assumed most of the samples in the data stream were normal these error rates are not very informative out of context. We can however, observe that Prophet performed best followed by GAN, DBSCAN and Welford, respectively. The results are consistent in all four scenarios. If we take into consideration that Prophet worked on the whole data set at once when the other three were limited to one sample at a time (as it is in production) we can declare that GAN performed best out of the algorithms that can detect anomalies on a live stream.

Algorithm	5770 Error rate	5771 Error rate	5772 Error rate	5773 Error rate
GAN	1.34%	1.38%	0.66%	1.09%
DBSCAN	1.59%	1.70%	1.78%	1.81%
Welford's algorithm	2.44%	2.41%	1.10%	2.31%
Facebook Prophet	1.14%	0.62%	0.39%	0.81%

Table 2: Error rates estimated from agreement rates for all four data streams.

We also considered a state-of-the-art method Isolation Forest, however it was too sensitive and therefore not usable in the error rate calculation.

5. CONCLUSIONS

We have tested five anomaly detection algorithms (Generative Adversarial Network, DBSCAN, Facebook Prophet, Welford's algorithm and Isolation Forest) on four separate data streams of water pressure data. Out of those five the Isolation Forest performed poorly since the share of anomalies found with this method was unreasonably high and was therefore not included in the final error estimates calculation.

Other approaches had similar shares of anomalies and were therefore used to calculate agreement rates and finally the estimated error rates of each anomaly detection algorithm. The results were consistent for all four data streams. Prophet performed best in every setting, however it looked at a data stream as a batch and it therefore could not be used for online anomaly detection. GAN performed second best followed by DBSCAN and Welford's algorithm which all work on a live data stream. Therefore we can conclude that the most fitting algorithm to be used for anomaly detection on the live water pressure data from water distribution network is GAN.

In future work, Facebook prophet could be adopted in such a way that it would also work on a live data stream since it has shown promising results in this experiment.

6. ACKNOWLEDGMENTS

This paper is supported by European Union's Horizon 2020 research and innovation programme under grant agreement No. 820985, project NAIADES (A holistic water ecosystem for digitisation of urban water sector).

7. REFERENCES

[1] Antonios Platanios, E. Estimating accuracy from unlabeled data.

[2] Ashrapov, I. Anomaly detection in time series with prophet library, Jun 2020.

[3] Celik, M., Dadaser-Celik, F., and Dokuz, A. S. Anomaly detection in temperature data using dbscan algorithm. *2011 International Symposium on INnovations in Intelligent SysTems and Applications* (2011).

[4] do Prado, K. S. How dbscan works and why should we use it?, Apr 2017.

[5] Ester, M., and Wittmann, R. Incremental generalization for mining in a data warehousing environment. In *International Conference on Extending Database Technology* (1998), Springer, pp. 135–149.

[6] Geiger, A., Cuesta-Infante, A., and Veeramachaneni, K. Adversarially learned anomaly detection for time series data, 2020.

[7] Kenda, K., and Mladenić, D. Autonomous sensor data cleaning in stream mining setting. *Business Systems Research: International journal of the Society for Advancing Innovation and Research in Economy 9*, 2 (2018), 69–79.

[8] Krieger, M. Time series analysis with facebook prophet: How it works and how to use it, Mar 2021.

[9] Lobo, J. L. Detecting real-time and unsupervised anomalies in streaming data: a starting point, Feb 2020.

[10] Schubert, E., Sander, J., Ester, M., Kriegel, H. P., and Xu, X. Dbscan revisited, revisited: why and how you should (still) use dbscan. *ACM Transactions on Database Systems (TODS) 42*, 3 (2017), 1–21.

[11] Taylor, S. J., and Letham, B. Forecasting at scale. *The American Statistician 72*, 1 (2018), 37–45.

[12] Thiyagarajan, K., Kodagoda, S., Ulapane, N., and Prasad, M. A temporal forecasting driven approach using facebook's prophet method for anomaly detection in sewer air temperature sensor system. In *2020 15th IEEE Conference on Industrial Electronics and Applications (ICIEA)* (2020), pp. 25–30.

Entropy for Time Series Forecasting

João Costa
Fakulteta za matematiko in fiziko
joaocostamat@gmail.com

António Costa
ESN Paris
antoniocbscosta@gmail.com

Klemen Kenda
Jožef Stefan Institut
klemen.kenda@ijs.si

João Pita Costa
IRCAI
joao.pitacosta@quintelligence.com

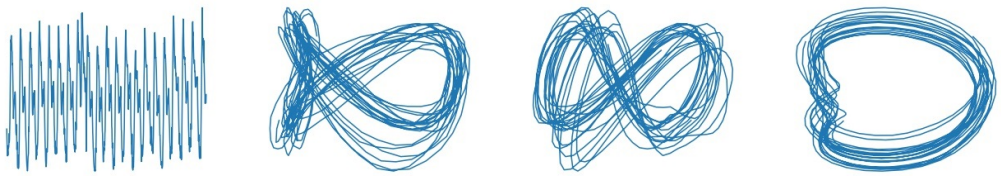

Figure 1: Sample of the time series and projections of the embedding - This plot gives us a geometrical representation of the theory involved in section 3 and shows the reconstructed state space of the given time series. This can be obtained by using Takens' embedding to reconstruct the time series y, given in figure a), as the markovian system Y_K with K time delays and then use Principal Component Analysis in order to perform the change of basis of the data. The obtained projections b), c) and d) attain the dynamics of the system, which gives us the possibility to predict the time series with higher efficiency.

ABSTRACT

In this paper, we present the exploitation of a method to extract information from microscopic samples of time series data in order to provide a representation of optimized stability to a chaotic system [1]. The main goal of this approach is to predict the dynamics of a time series and therefore develop optimized forecasting algorithms. First, we study how to increase the predictability of a system and second, we develop a Deep Learning Algorithm, namely an LSTM, that can recognize patterns in sequential data and accurately predict the future behaviour of a time series.

KEYWORDS

Recurrent Neural Networks, LSTM, Entropy, Markov Chain, Clustering, Time Series

1 INTRODUCTION

Given its intrinsic nature, mathematics concerns with the construction of formal statements and proofs relating the different concepts within it. Its methods are used in countless ways and effectively model the shape of our world. But how is it possible to shape the unknown? Motivated by this question and the upmost need for finding ways of optimizing water resources for future generations, there has been a great development on the study of dynamical systems based on, for example, (Shannon) entropy [9] and phase space reconstruction [4]. In this paper, we provide an approach to water resource management using Deep Learning and Chaos Theory, by studying the dynamics of a time series using the 2 main ideas cited before. This study was developed for the H2020 NAIADES Project [2] with data collected from the Municipality of Alicante (Spain). We will present this study for the Autobus Dataset, related to the Bus Station Areas in Alicante.

Information Society 2020, 5–9 October 2020, Ljubljana, Slovenia

2 STATIONARY AND CHAOTIC NATURE

2.1 Dickey-Fuller Test for Stationarity

In order to proceed with the theory involved in the method, it is necessary to understand the behaviour of the time series and its sensitivity to initial conditions. For studying time series' stationarity, one can use the Augmented Dickey-Fuller test, which is a type of statistical test called a unit root test, where generally the null hypothesis is that the time series can be represented by a unit root, which means that for $y = \{y_t\}_{t=1}^{T}$, the information at point y_{t-1} does not provide us the ability to predict y_t. In our case, we obtained that the p-value of the test was 0, so the null hypothesis was rejected and the time series has no unit root. Therefore, it is stationary and the time delays will provide important information for predicting the dynamics of the time series.

2.2 Lyapunov exponents for understanding chaotic nature

The Lyapunov Exponent is a quantifier for the sensitivity of the time series on initial conditions and therefore for its chaotic nature. The main idea is to select an array of nearest neighbors, i.e, points at minimum distance, and calculate its trajectories in time. By doing so, we can then obtain an average of this divergence exponent which gives us the Lyapunov Exponent. Since the system is bounded, the divergence is also bounded and will reach a plateau after a certain number of timesteps. In our case, the Lyapunov Exponent, given as the initial slope, is ≈ 518 and the initial growth is exponential, as can be seen in figure 5. Therefore, the time series is of a chaotic nature.

3 MAXIMUM PREDICTABILITY

Given the high variability of any chaotic system, it is hard to capture the whole set of variables that model the state space. This is characteristic of a non-Markovian system which is highly unpredictable. How do we surpass this issue?

Takens' Embedding Theorem [8] tells us that, under certain conditions, it is possible to use past data to reconstruct a Markovian system, thus giving us the possibility to model the initial time series with higher efficiency. We start by considering a set of ODEs $x = (\dot{x}_1, \dot{x}_2, \ldots, \dot{x}_D)$ and the d-dimensional time series $y(t)$ of duration T which is a set of incomplete measurements of x given by a measure M, i.e., $y = M(x)$. Then, in order to calculate the number of K time delays to feed the LSTM with, the d-dimensional measurements are lifted into the state space $Y_K \in \mathbb{R}^{d\times K}$ consisting of the previously referred K time delays [3]. It is possible to quantify the chaotic measure of the system Y_K by calculating the entropy resulting from clustering. This can be done by partitioning the $d \times K$-dimensional space into N Voronoi cells using K-Means clustering. Having partitioned the state space Y_K, the reconstructed dynamics are encoded as a row-stochastic transition probability matrix $P = [P_{ij}]_{i,j}$ which relates increments on the state-space density p in the following way

$$p_i(t + \delta t) = \sum_j P_{ji} p_j(t). \tag{1}$$

The entropy rate of the initial time series $y(t)$ is then approximated by estimating the entropy rate (Figure 3) of the associated Markov chain on the different time delays K using Kolmogorov's definition

$$h_{p_N}(K) = -\sum_{i,j} \pi_i P_{ij} \log P_{ij}, \tag{2}$$

where π is the estimated stationary distribution of the Markov chain P. This approximation gives an estimate for the conditional entropies (Figure 6), i.e., for a discrete state with delay vectors $\vec{y}^K = \{\vec{y}_i, \ldots, \vec{y}_{i+K-1}\}$, the entropy of the Markov chain provides an estimate for the conditional entropy,

$$\begin{aligned} h_{p_N}(K) &\approx \langle -\log[p_N(y_{i+K}|y_i, \ldots, y_{i+K-1})]\rangle \\ &= H_{K+1}(N) - H_K(N) \\ &= h_K(N), \end{aligned} \tag{3}$$

where H_K is the Shannon Entropy of the sequence obtained by partitioning the $\vec{y}$ space into N partitions.

4 MODEL ARCHITECTURE

4.1 LSTM

Long Short Term Memory (LSTM) Networks are a special type of Recurrent Neural Networks (RNN) which rely on gated cells that control the flow of information by choosing what elements of the sequence are passed on to the next module. This idea was introduced in order to surpass the vanishing gradient problem in conventional RNNs [7]. At each time t, consider f_t as the forget gate, i_t as the input gate and o_t as the output gate, which are functions that depend on the output of the previous LSTM module, given by h_{t-1} and on the input of the current timestep, given by x_t. Then, the next figure shows a representation of how a single LSTM cell performs its computations. The computations

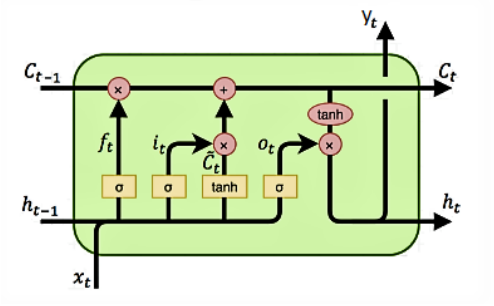

Figure 2: An LSTM performs the following ordered computations: The first step is to forget their irrelevant history. Then, LSTMs perform computation to decide on relevant parts of new information and based on the previous two steps, they selectively update the internal state. Finally, an output is generated.

shown in this figure can be mathematically represented as

$$\begin{aligned} f_t(x_t, h_{t-1}) &= \sigma(w_{f,x}^T x_t + w_{f,h} h_{t-1} + b_f) \\ i_t(x_t, h_{t-1}) &= \sigma(w_{i,x}^T x_t + w_{i,h} h_{t-1} + b_i) \\ o_t(x_t, h_{t-1}) &= \sigma(w_{o,x}^T x_t + w_{o,h} h_{t-1} + b_o), \end{aligned} \tag{4}$$

where $w_{f,x}, w_{i,x}, w_{o,x} \in \mathbb{R}^d$ are weight parameters and σ is an activation function.

4.2 Our approach

The core idea is to take a list of k training sets $Q_0, Q_1, \ldots, Q_{k-1}$ and testing sets $P_0, P_1, \ldots, P_{k-1}$ in order to generalize the model and do the best estimation for the time series. This is based on translating the testing sets' partitions along the time series, where the first partition $P_0 = \{p_0^0, \ldots, p_0^n\}$ is taken from the zeroth point of the time series data and the last partition $P_{k-1} = \{p_{k-1}^0, \ldots, p_{k-1}^n\}$ until the last point of the time series data and

$$|P_i| = \frac{|y|}{k}, \forall i \in \{0, \ldots, k-1\} \tag{5}$$

where $|y|$ stands for the cardinality of the time series y. This procedure yields k models which will use each of the training sets to make predictions on the respective test sets. Given the erratic nature of the data, which was taken in 15 and 30 minutes samples, a resampling to 30 minute delays had to be done on the 15 minutes delay data points and a masking was added to the time series in order to neglect NaN values that could be created from resampling. Therefore, a masking layer was added and the model is composed by 3 other layers $\mathcal{L}_{n_1}$, $\mathcal{L}_{n_2}$ and $\mathcal{L}_{n_3}$, where $n_1 = n_3 = 1$ (we have a univariate timeseries) and $n_2 = 64$, since it gave the best results in cross validation. A dropout regularization of 0.1 was added for better approximation of training and validation errors and the batch size was set to 128. The mean squared error for the predictions on the training set is ≈ 0.00115 and for the testing set is ≈ 0.00236. One can address the capacity of the model whose predictive results are shown in figure 4.

5 FORECASTING

5.1 Forecasting Methods

Consider a time series $T = \{t_1, \ldots, t_N\}$. The forecasting process can be done in 3 ways:

(1) iterated forecasting

(2) direct forecasting
(3) multi-neural network forecasting

Process number (1) is based on "many-to-one" forecast for which

$$t_{n+1} \approx \mathcal{F}(t_i, \ldots, t_{i+n-1}), i \in \{1, \ldots, N-n\}. \quad (6)$$

Then, a K-step forecast can be iteratively obtained by

$$\hat{t}_{N+j} := \mathcal{F}(\hat{t}_{N+j-n+1}, \ldots, \hat{t}_{N+j-2}, \hat{t}_{N+j-1}), j \in 1, \ldots, K. \quad (7)$$

Process number (2) can be characterized by training a "many-to-many" function $\mathcal{F}$ for which

$$(t_{i+n}, \ldots, t_{i+n+K-1}) \approx \mathcal{F}(t_i, \ldots, t_{i+n-1}), \quad (8)$$

where $i \in \{1, \ldots, N-n-K+1\}$. We can obtain a K-step forecast by

$$(\hat{t}_{N+1}, \ldots, \hat{t}_{N+K}) := \mathcal{F}(t_{N-n+1}, \ldots, t_N). \quad (9)$$

Finally, process (3) is defined by k "many-to-one" functions $\mathcal{F}_1, \ldots, \mathcal{F}_k$ which hold the following relationship

$$\begin{aligned} t_{i+n} &\approx \mathcal{F}_1(t_i, \ldots, t_{i+n-1}) \\ &\vdots \\ t_{i+n+K-1} &\approx \mathcal{F}_k(t_i, \ldots, t_{i+n-1}), \end{aligned} \quad (10)$$

where i ranges from 1 to $N-n-K+1$. Process (1) does not require k a propri while both process (2) and (3) are dependent on the choice of k.

5.2 Our Approach

We chose to do a Direct Forecasting for the next 7 days by taking the last test set partition P_{k-1} and did a prediction on this test set. Although forecasting seems pretty motivating, by choosing a partition that attains more characteristics of the time series, one can achieve even better results. The achieved forecast can be seen on Figure 8 and compared with a 7 days sample on Figure 7.

6 RESEARCH METHODS

6.1 Time Series Reconstruction

Consider the time series y with duration T as given in section 2. The idea is to add K time delays to y in order to obtain a $(t-K) \times Kd$ space $Y_K \in \mathbb{R}^{d \times K}$ and further partition Y_K using k-means Clustering into N Voronoi Cells.

6.2 Entropy Calculation

Consider the N Voronoi Cells given as the number of partitions of Y_K and consider the joint probability $p(c_{i_1}, \ldots, c_{i_l}), \{i_1, \ldots, i_l\} \in \{0, \ldots, N-1\}$. Then, the Shannon Entropy [6] is given by

$$H_l = -\sum p(c_{i_1}, \ldots, c_{i_l}) \log p(c_{i_1}, \ldots, c_{i_l}) \quad (11)$$

and the conditional probabilites are given by

$$p(c_{i_{l+1}} | c_{i_1}, \ldots, c_{i_l}), \quad (12)$$

where ci_{l+1} is the next Voronoi Cell after c_{i_l}. We can calculate the entropy rate growth by considering the conditional probabilities of the system given the previous l cells, when visiting the $(l+1)$-th cell, via

$$h_l = \langle -\log[p(c_{i_{l+1}} | c_{i_1}, \ldots, c_{i_l})] \rangle = H_{l+1} - H_l \quad (13)$$

Taking the supremum limit over all possible partitions P of Y_K, we obtain the Kolmogorov-Sinai invariant of the system,

$$h_{KS} = \sup_P \lim_{l \to \infty} h_l(P). \quad (14)$$

6.3 Data and Code Git Repository

The complete work can be found in:
https://github.com/johncoost/JoaoModelsForAlicante.

7 PLOT OF RESULTS

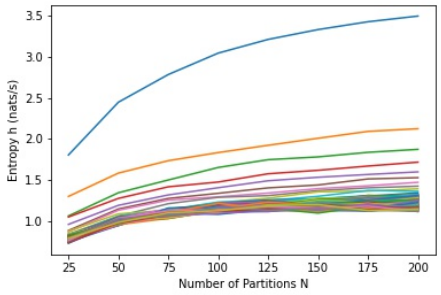

Figure 3: Entropy Rate h - The entropy rate h is given as the function of the number of partitions N for increasing number of delays K (given by the different colors in a descendent mode). It is possible to observe that the entropy rate is a non-decreasing function on the number of partitions N. The idea is to choose the value of N for which the entropy is maximum so that we have the maximum possible information about the system's dynamics.

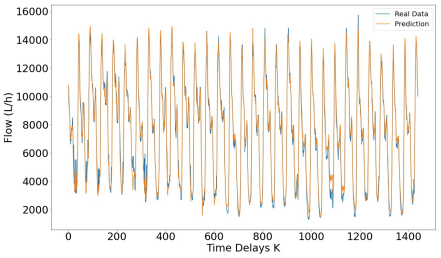

Figure 4: Prediction on the last test set - This shows a sample of the last test set and its prediction. We can observe the effectiveness of the LSTM in modelling the given time series by having a deep understanding of its inherent dynamics.

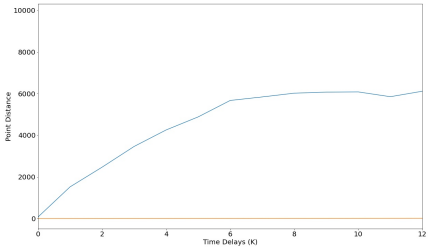

Figure 5: In this figure, we can understand the initial exponential growth on distance between points (given in blue), relative to a curve of slope 1 (given in orange).

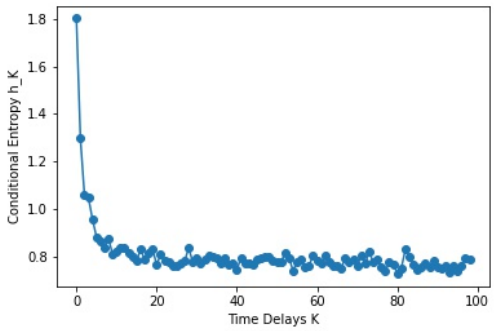

Figure 6: Conditional Entropies - In this plot we can see the entropy rate for number of partitions $N = 200$ which maximizes this entropy. This function reaches a plateau at ≈ 24 timesteps, which gives us an idea about which is the optimal K to choose. Given that we have 30 minutes timesteps, this plot shows that the optimized time delay is of 12h which corresponds to the day and night cycles

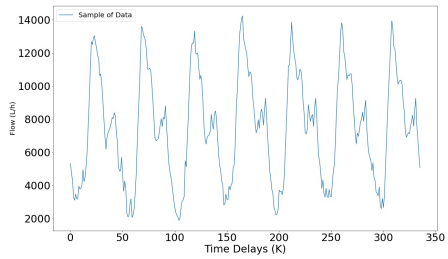

Figure 7: 7 Days Sample

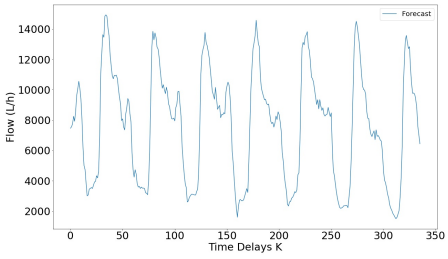

Figure 8: Prediction for 7 days ahead - Actual forecast using 336 timesteps that gives a 7 day future forecast sample using the LSTM model and direct forecasting. It is possible to observe that, as in figure 6, the values vary between ≈ 2000 to ≈ 14000 flow units and the essential dynamics of the time series were understood by the LSTM.

8 CONCLUSION

Having developed all the necessary machinery for constructing a coherent forecasting engine, we come to the conclusion that although the cardinality of the time series data was relatively small, the obtained results are promising and the model will certainly show satisfying results when applied in real time.
For the future, we want to continue developing the project by building other algorithms, such as Transformer neural network, that would provide even better results. Another idea is to use weather data and build a multivariate LSTM that optimally gives better results than the univariate one.

9 ACKNOWLEDGMENTS

I greatly thank to António Carlos Costa for working in cooperation and giving me the possibility to use the powerful machinery he built in order to obtain the desired K time delays and understand the complex dynamics of the system. Also, to the NAIADES team at Jožef Stefan Institute for all the knowledge exchange and, in particular, to Klemen Kenda for giving me the possibility of writing this paper and João Pita Costa for giving me insights on how to write and structure the paper.

This paper is supported by European Union's Horizon 2020 research and innovation programme under grant agreement No. 820985, project NAIADES (A holistic water ecosystem for digitisation of urban water sector).

REFERENCES

[1] Tosif Ahamed, Antonio Carlos Costa, and Greg J. Stephens. 2019. Capturing the continuous complexity of behavior in c. elegans. (2019). arXiv: 1911.10559 [q-bio.NC].

[2] 2019-2022. Cordis, "naiades project". In CORDIS. \url{https://cordis.europa.eu/project/id/820985}.

[3] Antonio Carlos Costa, Tosif Ahamed, David Jordan, and Greg Stephens. 2021. Maximally predictive ensemble dynamics from data. (2021). arXiv: 2105.12811 [physics.bio-ph].

[4] Vicente de P. Rodrigues da Silva, Adelgicio F. Belo Filho, Vijay P. Singh, Rafaela S. Rodrigues Almeida, Bernardo B. da Silva, Inajá F. de Sousa, and Romildo Morant de Holanda. 2017. Entropy theory for analysing water resources in northeastern region of brazil. *Hydrological Sciences Journal*, 62, 7, 1029–1038. DOI: 10.1080/02626667.2015.1099789. eprint: https://doi.org/10.1080/02626667.2015.1099789. https://doi.org/10.1080/02626667.2015.1099789.

[5] David A. Dickey and Wayne A. Fuller. 1979. Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American Statistical Association*, 74, 366a, 427–431. DOI: 10.1080/01621459.1979.10482531. eprint: https://doi.org/10.1080/01621459.1979.10482531. https://doi.org/10.1080/01621459.1979.10482531.

[6] Robert M. Gray. 2011. *Entropy and Information Theory*. (2nd edition). Springer Publishing Company, Incorporated. ISBN: 9781441979698.

[7] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9, 8, 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

[8] Floris Takens. 1981. Detecting strange attractors in turbulence. In *Lecture Notes in Mathematics*. Springer Berlin Heidelberg, 366–381. DOI: 10.1007/bfb0091924.

[9] Peyman Yousefi, Gregory Courtice, Gholamreza Naser, and Hadi Mohammadi. 2020. Nonlinear dynamic modeling of urban water consumption using chaotic approach (case study: city of kelowna). *Water*, 12, 3. ISSN: 2073-4441. DOI: 10.3390/w12030753. https://www.mdpi.com/2073-4441/12/3/753.

Modeling stochastic processes by simultaneous optimization of latent representation and target variable

Jakob Jelenčič
Jozef Stefan International Postgraduate School
Jozef Stefan Institute
Ljubljana, Slovenia
jakob.jelencic@ijs.si

Dunja Mladenić
Jozef Stefan International Postgraduate School
Jozef Stefan Institute
Ljubljana, Slovenia
dunja.mladenic@ijs.si

ABSTRACT

This paper proposes a novel method for modeling stochastic processes, which are known to be notoriously hard to predict accurately. State of the art methods quickly overfit and create big differences between train and test datasets. We present a method based on simultaneous optimization of latent representation and the target variable that is capable of dealing with stochastic processes and to some extent reduces the overfitting. We evaluate the method on equities and cryptocurrency datasets, specifically chosen for their chaotic and unpredictable nature. We show that with our method we significantly reduce overfitting and increase performance, compared to several commonly used machine learning algorithms: Random forest, General linear model and LSTM deep learning model.

1. INTRODUCTION

Time series prediction has always been an interesting challenge. Deep learning structures that are designed for time series are prone to overfitting. Especially if the underlying time series is stochastic by nature. Every young researcher's first attempt when dealing with time series, was trying to learn a time series model that will predict future prices; whether in equities, commodities, forex or cryptocurrencies. Unfortunately it is not that simple. One can easily build a near perfect model on the train dataset just to find it is completely useless on the test dataset.

We propose a novel method that is capable of effectively combatting the overfitting, especially this proves to be a difficult task when one is dealing with a problem directly applicable in practical situations. The main idea is to add noise from the same distribution as the training data and then at the same time optimize the target variable and the latent representation with the help of the autoencoder. The longer the training goes, the lower is the amplitude of noise and the less focus is on the optimization of the representation.

We have evaluated the proposed method on an equities dataset and a cryptocurrency dataset, in both cases achieving extraordinary results on the test dataset. We have also shown the importance of noise distribution and how the de-noising fails if the distributions of the data and noise do not align.

The rest of the paper is organised as follows. Section 2 describes the data we were using. In section 3 we introduce the proposed method. In section 4 we present empirical results. In section 5 we conclude by pointing out the main results and defining guidance for the future work.

2. DATA

The proposed method works well for stochastic processes. Equities are supposed to follow some form of stochastic process [9], either the Black-Scholes one or some more complex process with unknown formulation. In order to evaluate our method, we have collected daily data of more than 5000 equities listed on NASDAQ from 2007 on. The data is freely available on the Yahoo Finance website [2]. We transformed the data using technical analysis [10] and for test set took every instance that happened after 2019. We calculated moving average using 10 days closing price then tried to predict the direction of the change of this trendline.

The equity data turned out to be a little bit timid, not chaotic enough to demonstrate the full ability of the proposed method. This is why we also collected minute data of cryptocurrencies Ethereum and Bitcoin and used the method on them as well. Data is available on the crypto exchange Kraken [1]. We used the same transformation as for the equities, but with a bit quicker trend. This time the target variable was change in the trendline in the next 6 hours. For the test set we took every instance that has time stamp after December 2020.

The reader should note that the end goal is not to accurately predict future equity price, since that is next to impossible. As soon there is a pattern, someone will profit from it and then the pattern will change. By predicting the future trend line, one can obtain a significant confidence interval and estimates of where the price could be, and then design for example a derivative strategy that searches for favourable risk versus rewards trades.

3. PROPOSED METHOD

We propose the method designed for prediction of stochastic processes. The method achieves significant results improving the metrics and loss functions on unseen data, where standard deep learning is prone to over-fit. The main advantage is reducing the gap between training data and testing data, sometimes to a degree where one sacrifices a little bit on the train side to actually have the model outperforming it on test data. This is very important in time series, where a prediction model is usually just one part of a bigger strategy and where the train over-fit is the biggest issue. For example, designing a trading strategy on over-fitted predictions, that kind of mistake can lead to huge capital losses.

The proposed method can be broken down into 3 important parts: normalization, noise addition and additional optimization of latent representation. Each part can be easily integrated into an already existing pipeline.

3.1 Empirical normalization

Normalization plays an important role in deep learning models. It was shown that normalization significantly speeds up the gradient descent, almost independently of where normalization takes place. It can be weight normalization [11] during the actual optimization, or it can be the batch normalization [8], or just normalization of the whole input data [7]. In the proposed method it is important that the 3 dimensional input data comes from the same distribution as the generated noise. Since it is fairly straightforward to sample data from a 3 dimensional normal distribution, we normalize input data using an empirical cumulative distribution function [12] and empirical copula [4] [5]. We align all central moments of the unknown distribution to the ones from centered and standardised normal distribution. The normalization takes place before the data is reshaped to 3 dimensional tensor.

3.2 Noise addition

Introduction of the noise is not new in unsupervised learning and it was shown that it has a positive effect [14]. Adding noise to input data and then forcing the model to learn how to ignore it has a lot of success in generative adversarial networks [3], where convergence can be very tricky to achieve. We transformed that idea and embedded it into supervised learning procedure. The noise addition is described in Algorithm 1.

In Algorithm 1 we will use the following abbreviations.

- $X = [bs, ts, np]$ stands for the input tensor with 3 dimensions; batch size, time steps and number of features used for predictions.
- α, β are parameters that control how fast noise will decrease during the training procedure. They should be between 0 and 1, where lower value correspond to a faster decrease in the amplitude of the added noise.
- mvn stands for function sampling from a two dimensional correlated Gaussian distribution, where Σ is the covariance. $matmul$ stands for matrix multiplication.

Algorithm 1 Noise definition

Inputs: X, α, β, $epoch$
$Y = [ts, ts, np]$ ▷ Array for holding Cholesky decompositions of time correlation matrices.
for $t \in \{1, \ldots, np\}$ **do**
  $\Sigma_t = cov(X[, , t])$
  $Y[, , t] = chol(\Sigma_t)$ ▷ In practice the closest positive definite matrix of Σ_t is computed before the Cholesky decomposition.
end for
$Z = [bs, ts, np]$ ▷ Array for holding noise samples.
for $i \in \{1, \ldots, ts\}$ **do**
  $\Sigma_i = cov(X[, i,])$
  $Z[, i,] = mvn(bs, \Sigma_i)$
end for
for $j \in \{1, \ldots, np\}$ **do**
  $Z[, , j] = matmul(Z[, , j], Y[, , j])$ ▷ Correcting initially independent noise samples with respect to time.
end for
for $w \in \{1, \ldots, ts\}$ **do**
  $Z[, w,] = Z[, w,] * ((\beta^{ts-w} \cdot \alpha^{epoch}) \cdot sd)$ ▷ Decrease the noise during the training procedure.
end for
$R = X + Z$
Return R.

3.3 Optimization of latent representation

The most common issue with deep learning optimization is falling into a local optimum and being unable to move past it [13]. We introduce autoencoder part into the optimization procedure in order to force the model to shift from going directly to local optimum to learning the latent representation first. We expect that this combined with the addition of noise, will force the model first to learn how to ignore the noise that we added and the noise that is already in the data by nature of the stochastic process [15]. We optimized the model using the Adam optimizer [6]. The loss function used in optimization is defined like:

$$L = L_Y + W_{ae} \cdot decay^{epoch} \cdot L_{ae},$$

where L_Y stands for the supervised loss function which will depend on the problem while L_{ae} stands for the loss between encoded output and input data. Decay weight is decreasing the longer the training goes on.

4. RESULTS

We have divided the results section into 2 parts: unsupervised and supervised. In the first we demonstrate why the noise distribution is important. For the unsupervised part, due to hardware constraints, we have only used the cryptocurrency dataset since we deemed it more demanding than the equity one. In the second, we demonstrate how the our method increases test metric on both datasets.

4.1 Unsupervised learning results

In order to test the efficiency of distributed noise versus just random noise, we created 3 models. The baseline model was a deep learning model with 3 stacked LSTM layers, encoded layer, then again 3 stacked LSTM for decoded output. We have used Adam as optimizer. As loss function we used mean-squared error. We have stopped the learning after there was no improvement for 25 epochs on the validation set. The validation set was randomly taken out of the train set. Parameters α and β were both set to 0.99 and sd was initially set to 1.25. The noise decreases with learning procedure. Interestingly keeping noise constant did not achieve any results.

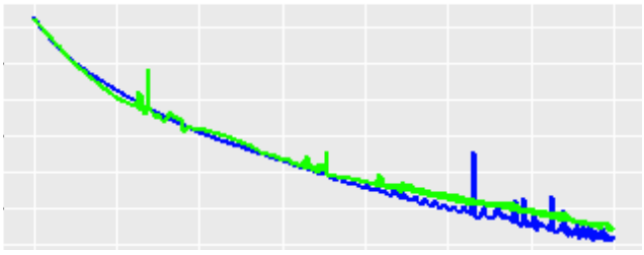

Figure 1: Test loss of autoencoder model with random noise (green) versus no noise (blue).

Initially we have tested baseline model versus de-noising model but with uncorrelated noise. In the Figure 1 is plotted the de-noising test loss function in green colour and the baseline test loss function in blue. Training was stopped relatively early compared to Figure 2 and it is also obvious that de-noising test loss is even worse than that of the classic autoencoder.

In the second example we switched from uncorrelated noise to the noise with same distribution as input data. As is apparent on Figure 2, where again we have de-noising test loss plotted with green and classic test loss with blue, the de-noising autoencoder achieved lower test loss than the classic one.

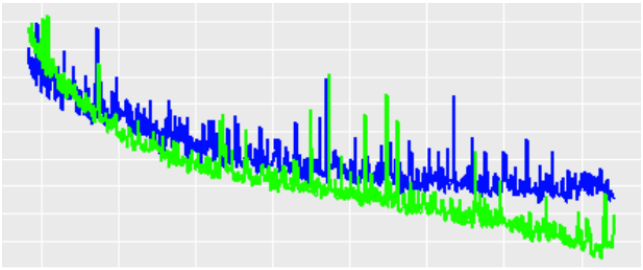

Figure 2: Test loss of autoencoder model with correlated noise (green) versus no noise (blue).

What we expected is that then the train and validation losses will be worse than with the classic autoencoder. Surprisingly, that was not the case. With the de-noising autoencoder using noise with the same distribution as the input data, both train and validation losses were better than with classic one. This result is definitely worth further investigation and experimentation.

4.2 Supervised learning results

In the previous section we have shown that the distribution of the noise matters. In this section we will show that noise combined with optimization of latent representation significantly improves metrics on unseen data. Similarly as before, α and β were both set to 0.99 and sd was initially set to 1.25. From our experience this setting achieves the best results, but further exploration needs to be done. W_{ae} was initially set to 5 and $decay$ to 0.95.

Since we now operate in a supervised environment, we can compare our models to the majority class. But to really demonstrate the effectiveness of the method, we chose to compare the following models:

- Majority class, which serves as a sanity check.
- Random Forest with 500 trees.
- Generalized linear model.
- Deep learning model with 3 stacked LSTM layers.
- Deep learning model with 3 stacked LSTM layers and optimization of latent representation.
- Deep learning model with 3 stacked LSTM layers and correlated noise addition.
- Finally, deep learning model with 3 stacked LSTM layers and correlated noise addition and optimization of latent representation.

All 4 of the deep learning models are identical, all are optimized with Adam and categorical cross entropy was used as a loss function for the supervised part and mean squared error for the autoencoder part. Initially we have only tested the models on equities data, but it turned out that the equities were not chaotic enough. By that we mean that especially with deep learning models the difference between train and test loss was not so big that it would be problematic. From previous work experience we know that overfit is a big issue in cryptocurrency dataset, so then we decided to test that dataset in a supervised setting as well. All models were trained three times on each dataset and the results in Table 1 and Table 2 are the averages of the 3 runs.

In Table 1 we show the results from the equity dataset. Our method managed to improve test accuracy (from 0.673 to 0.682) without decreasing train accuracy (0.681). Maintaining test accuracy and keeping it comparable to test one is important if one needs to build additional strategy upon predictions. Just noise addition slightly improved the results (from 0.673 to 0.675), while just the optimization of the latent distribution does not improve anything.

Table 1: Supervised results on equity dataset.

Method	Train Accuracy	Test Accuracy
Majority	0.513	0.537
Random Forest	0.649	0.655
GLM	0.664	0.655
LSTM	0.681	0.673
latent LSTM	0.633	0.673
noise LSTM	0.681	0.675
latent noise LSTM	0.681	0.682

In Table 2 we show results from the cryptocurrency dataset. Similar as on the equity dataset, our method behaves as intended on the cryptocurrency dataset as well. We can see reduced overfitting that is apparent in the normal LSTM model. With those results we can conclude that the proof of concept works, but for additional claims we will need more testing and deeper parameter analysis.

Table 2: Supervised results on cryptocurrency dataset.

Method	Train Accuracy	Test Accuracy
Majority	0.512	0.556
Random Forest	0.689	0.692
GLM	0.682	0.695
LSTM	0.754	0.696
latent LSTM	0.736	0.683
noise LSTM	0.697	0.695
latent noise LSTM	0.706	0.714

It is interesting to point out that with the proposed method the test loss on cryptocurrency dataset was 0.552, while train loss was 0.592. While 0.552 was the best loss any deep learning model achieved, that wide difference indicates that we could improve our model even further by fine tuning the parameters.

5. CONCLUSIONS AND FUTURE WORK

In this work we have introduced and demonstrated how the addition of noise and simultaneous optimization of latent representation and target variable reduce overfitting on time series data. In the unsupervised case we have shown that the distribution of the noise matters and the input data must align to achieve maximum effect from the noise addition.

In the future work we have to estimate the effect of the newly introduced parameters on method's convergence. At the same time we need to explore how the method behaves when embedded into larger models, transformers for example. We also need to evaluate the method in datasets that are by nature stochastic but do not come from the financial domain. Finally, we need to evaluate our method on a dataset that is not stochastic.

6. ACKNOWLEDGMENTS

This work was supported by the Slovenian Research Agency. We also wish to thank prof. dr. Ljupčo Todorovski for his help, especially with unsupervised results.

7. REFERENCES

[1] *Kraken exchange*. `https://www.kraken.com/`.

[2] *Yahoo Finance*. `https://finance.yahoo.com/`.

[3] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65, 2018.

[4] P. Jaworski, F. Durante, W. K. Hardle, and T. Rychlik. *Copula theory and its applications*, volume 198. Springer, 2010.

[5] H. Joe. *Dependence Modeling with Copulas*. CRC Press, 2014.

[6] D. Kingma and J. Ba. *Adam: A Method for Stochastic Optimization*. 2014. `https://arxiv.org/abs/1412.6980`.

[7] K. Y. Levy. The power of normalization: Faster evasion of saddle points. *arXiv preprint arXiv:1611.04831*, 2016.

[8] M. Liu, W. Wu, Z. Gu, Z. Yu, F. Qi, and Y. Li. Deep learning based on batch normalization for p300 signal detection. *Neurocomputing*, 275:288–297, 2018.

[9] R. C. Merton. Option pricing when underlying stock returns are discontinuous. *Journal of financial economics*, 3(1-2):125–144, 1976.

[10] J. J. Murphy. *Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications*. New York Institute of Finance Series. New York Institute of Finance, 1999.

[11] T. Salimans and D. P. Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. *Advances in neural information processing systems*, 29:901–909, 2016.

[12] B. W. Turnbull. The empirical distribution function with arbitrarily grouped, censored and truncated data. *Journal of the Royal Statistical Society: Series B (Methodological)*, 38(3):290–295, 1976.

[13] R. Vidal, J. Bruna, R. Giryes, and S. Soatto. Mathematics of deep learning. *arXiv preprint arXiv:1712.04741*, 2017.

[14] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103, 2008.

[15] N. Wax. *Selected papers on noise and stochastic processes*. Courier Dover Publications, 1954.

Causal relationships among global indicators

Matej Neumann
Jožef Stefan Institute
Jamova cesta 39, Ljubljana, Slovenia
matej.neumann@student.fmf.uni-lj.si

Marko Grobelnik
Jožef Stefan Institute
Jamova cesta 39, Ljubljana, Slovenia
marko.grobelnik@ijs.si

ABSTRACT

It is important to know how changing one thing will affect another. This becomes even more important when the thing we are changing will affect a lot of people. Therefore, we need a way to visualize how all the things are connected. In this paper, we will demonstrate an approach that uses Granger causality to find causal relationships between global indicators. Our results show that global indicators are indeed highly interconnected however, they still need to be looked at within each country individually. We also comment how this approach can be used to help with policy making decisions.

KEYWORDS

Causality, Global indicators, Granger, Timeseries, SDGs

1 INTRODUCTION

The Sustainable Development Goals (SDGs) launched on January 1, 2016 include 17 goals, 169 targets and 232 unique indicators with the intent to help frame the policies of the United Nations' (UN) member states through 2030 [8]. Because the goals are highly interconnected, as the indicators are not independent, it is important to understand synergies, conflicts and causal relationships between them to support decisions. Without such understanding a policy to help one goal could hurt another. For example, a policy aiming to improve hunger could conflict with climate-mitigation. This paper will focus on finding such relationship with Granger causality.

Granger causality is a statistical concept of causality that is based on prediction and was traditionally only used in the financial domain however, over recent years there has been growing interest in the use of Granger causality to identify causal interactions in neural data [6].

Similar works such as [7] and [2] have already looked for causal relationships between specific SDGs. This paper confirms the previously done work and expands it by adding additional indicators and looking for causal relationship between all the indicators, not just the ones focused on SDGs.

In paper [2] the authors say that the analysis of all of the indicators country by country is without doubt impractical. Nevertheless, Table 2 shows that however impractical it may be, it is still required, as even neighboring countries have vastly different causal relationships.

2 DESCRIPTION OF DATA

2.1 United Nations Statistics Division (UNSD)

This is the official source published by the United Nations it provides information on the development and implementation of an indicator framework for the follow up and review of the 2030 Agenda for Sustainable Development [4].

2.2 The World Bank (WB)

As the data set provided by the UN itself often has missing values, which results in unhealthy timeseries and unreliable results, we decided to add the dataset "World Development Indicators" from The World Bank [5]. Although the data set might not be as official as the one provided by the UN, it does contain 1440 unique indicators for 266 different countries and groups, where each indicator contains a timeseries ranging from the year 1960 to the present time. This addition does not only make the dataset healthier, it also introduces new indicators that are not listed in the UN SDGs. Even so our new dataset still has some limitations. From Figure 1 we can see that on average a country or groups has no values for around 33% of its indicators. Therefore, from now on when talking about the indicators, we will restrict ourselves to just those ones that have at least 20 nonmissing values in their timeseries. This restriction will insure that we are always dealing with a healthy timeseries and it is justified as on average those indicators make up about 50% of all of the ones available as seen in Figure 2.

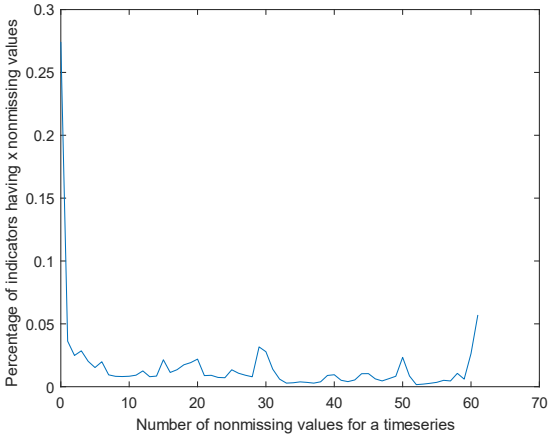

Figure 1: Percentage of indicators having x nonmissing values in its timeseries.

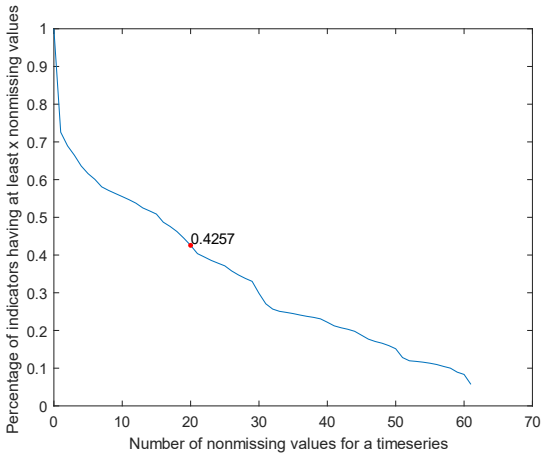

Figure 2: percentage of indicators having at least x nonmissing values in its timeseries.

To better imagine what kind of indicators we are dealing with, we can check Table 1 which shows the top 10 most common ones.

Indicator name	Frequency
Renewable electricity output (% of total electricity output)	265
Population, total	265
Population growth (annual %)	265
Nitrous oxide emissions in energy sector (thousand metric tons of CO2 equivalent)	265
Methane emissions in energy sector (thousand metric tons of CO2 equivalent)	265
Agricultural nitrous oxide emissions (thousand metric tons of CO2 equivalent)	265
Agricultural methane emissions (thousand metric tons of CO2 equivalent)	265
Urban population growth (annual %)	263
Urban population (% of total population)	263
Urban population	263

Table 1: Most common indicators and their frequency of 20 nonmissing values

3 METHODOLOGY

3.1 Granger causality

The causal relationships between indicators were determined by the Granger causality test. The Granger causality test is a statistical hypothesis test for determining whether one timeseries is useful in forecasting another. Informally we say that timeseries X Granger-causes timeseries Y if predictions of the value of Y based on its own past values and on the past values of X are better than predictions of Y based only on Y's own past values. Or in other words X Granger-causes Y if we can better explain the future values of Y with both the past values of X and Y and not just the past values of Y.

More formally, let x and y be stationary timeseries and let x(t) and y(t) be the univariate autoregression of x and y respectfully:

$$x(t) = b_0 + \sum_{i=1}^{p} b_i x(t-i) + E_2(t)$$

$$y(t) = a_0 + \sum_{i=1}^{p} a_i y(t-i) + E_1(t)$$

where p is the number of chosen lagged values included in the model, a_i and b_i are contributions of each lagged observation to the predicted values of $x(t)$ and $y(t)$ and $E_i(t)$ the difference between the predicted value and the actual value. To test the null hypothesis that x does not Granger-cause y, we augment $y(t)$ by including the lagged values of x to get:

$$y(t) = c_0 + \sum_{i=1}^{p} a_i y(t-i) + b_i x(t) + E_3(t).$$

We then say that x Granger-causes y if the coefficients b_i are jointly significantly different from zero. This can be tested by performing an F-test of the null hypothesis that $b_i = 0$ for all i.

3.2 Statistical significance and the p-value

In testing, a result has statistical significance if it is unlikely to occur assuming the null hypothesis. More precisely, a significance level α, is the probability of the test rejecting the null hypothesis, given that the null hypothesis was assumed to be true and the p-value is the probability of getting result at least as extreme, given that the null hypothesis is true. Then we say that the result is statistically significant when $p \leq \alpha$.

3.3 Limitations of the Granger causality test

As its name implies, Granger causality is not necessarily true causality. Having said this, it has been argued that given a probabilistic view of causation, Granger causality can be considered true causality in that sense, especially when Reichenbach's "screening off" notion of probabilistic causation is considered [1].

A problem may occur if both timeseries x and y are connected via a third timeseries z. In that case our test can reject the null hypothesis even if manipulation of one of the timeseries would not change the other. Other possible sources of problems can happen due to: (1) not frequent enough or too frequent sampling, (2) time series nonstationarity, (3) nonlinear causal relationship.

4 EXPERIMENTS

4.1 Setup

Due to time constraints and the limitations of my home system, we decided to limit ourselves to taking just a few countries and groups and calculating the causality relationships for them. The ones we decided on are: (1) United States, (2) China, (3) Uruguay, (4) Slovenia, (5) Austria, (6) Croatia, (7) Italy, (8) European Union and (9) OECD. Our plan was to choose

	AUS	CH	CRO	EU	ITA	OECD	SLO	UY	USA
AUS	100%	4.8%	5.1%	6.9%	6.7%	6.0%	5.9%	4.4%	7.1%
CH		100%	5.6	3.5%	4.3%	3.9%	4.2%	4.7%	4.3%
CRO			100%	4.6%	5%	3.3%	6.6%	3.8%	5.6%
EU				100%	11%	20%	5.7%	3.6%	10%
ITA					100%	6.7%	7.5%	3.8%	6.7%
OECD						100%	5%	3%	17%
SLO							100%	3.5%	5.6%
UY								100%	4.2%
USA									100%

Table 2: Percentage of same causal relationships.

a few of the major world powers and compare the differences and similarities between the causal relationships.

4.2 Modeling the dataset

Once the data was collected from the UNSD and WB website it first had to be put into a suitable form. We decided on a 3D matrix where the first component represented the country or group, the second component represented the time series and last one representing the indicator.

4.3 Parameters

As mentioned before, when searching for causal relationships in a certain country or group we limit ourselves only to those indicators who have at least 20 nonmissing values. Furthermore, we chose a significance level of 0.05 or 5% and tested for lagged values from 1 to 4.

4.4 Determining causality

Once the modeling was done and the parameters were set we first needed to make sure that the timeseries were stationary. To do that we ran the ADF-test and differenced the times series accordingly to make them stationary. Then we ran the Granger-causality test 4 times, once for each lagged value, for each of the 9 countries and groups listed in 4.1. The results for each lagged value were then saved in a 1440x1440 weighted adjacency matrix, where the (i,j) element was nonzero if and only if the i-th indicator Granger-caused the j-th indicator for all lagged values between 1 and 4 and had the weight of the average of the 4 p-values.

Once we had the weighed adjacency matrix we matched the available indicators with the 17 SDGs by comparing the most common buzzwords found in the description of the SDGs and the name of the indicators. An example of some of the buzzwords can be seen in Table 3.

5 RESULTS

With the weighted adjacency matrix in hand, it is sensible to ask ourselves whether there exist any causal relationships that hold true for each of the tested countries or groups. The answer is positive as seen in Figure 3. We can however see that the only causal relationships that survived were the ones that connected different population ages to each other. This result seems sensible as in general no two countries are exactly the same and are therefore going to have a unique set of causal relationships. That being said one can easily imagine why each population age Granger-causes the next one. For example, if we know the percentage of people aged 4, we can pretty accurately predict what the percentage of people aged 5 is going to be in the next year.

SDG	Buzzwords
Zero Hunger	nourishment, food, stun, anemia, agriculture
Clean Water and Sanitation	water, sanitation, drinking, drink, hygiene, freshwater
Affordable and Clean Energy	energy, electricity, fuel
Climate Action	disaster, disasters, climate, natural, risk, Sendai, environment, environmental, green, developed, pollution
Good Health and Well-Being	mortality, birth, infection, tuberculosis, malaria, hepatitis, disease, cancer, diabetes, treatment, Alcohol, death, birth, health, pollution, medicine

Table 3: Some of the most common buzzwords found in SDGs

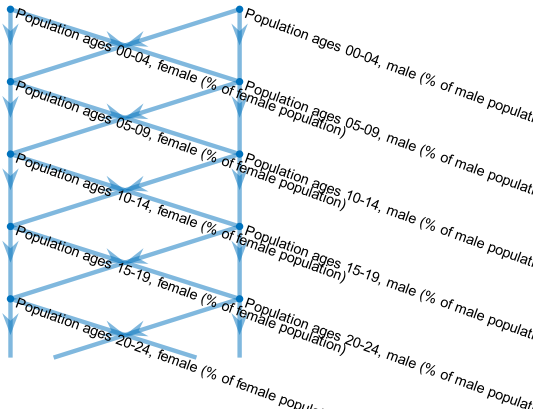

Figure 3 Only causal relationships that are true for each of the 9 countries and groups (continuous down).

On the other hand, one may assume that if we compare countries which are close to each other or are historically connected then the causal relationships should not differ by a lot.

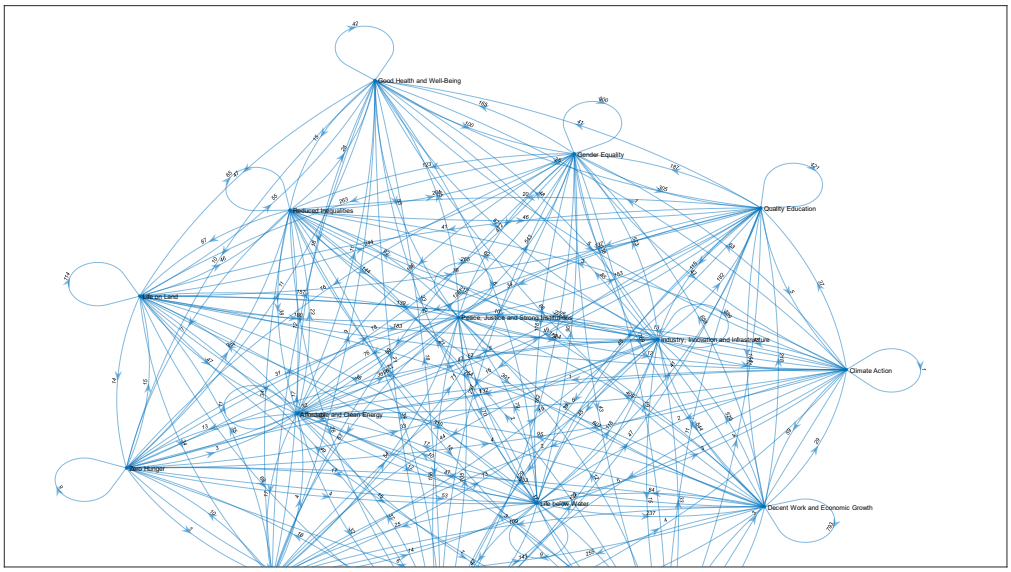

Figure 4: Interconnectedness of SDGs.

That however is not the case as can be seen in Table 2. This suggests, that when talking about causal relationships, one must look at each country or group individually.

Therefore, let's focus just on Slovenia. Due to Slovenia having 10083 positive causal relationships we will limit ourselves to just those that interact with SDGs. Figure 4 shows that indeed SDGs are not independent and in fact are highly interconnected. The presence of self-loops also suggests that there exist causal relationships between indicators of an SDG itself. This result has two consequences:

- When thinking about policies aiming to improve one goal we need to be careful to not harm another
- Instead of outright improving one goal, we can instead focus the ones that are in causal relationship with the one we wish to improve

Let's give an example. Suppose we would want to implement a policy to help to help lower the suicide mortality rate, but we are not how to do that directly. We can therefore instead check which indicators Granger-cause the one we are trying to improve. In our case the indicator "Unemployment, youth total (% of total labor force ages 15-24)" Granger-causes the suicide mortality rate. Therefore, if we improved the % of unemployed young people we would be able to also reduce the suicide mortality rate which was our initial goal.

6 CONCLUSION AND FUTURE WORK

In this paper we demonstrated an approach for calculating causality between depending global indicators and mentioned how this can help with implementing policies. We also showed that neighboring and similar countries in general don't have the same causal relationships, which makes it hard to group them together. However, finding such a grouping, if it exists, could be done in the future. The approach shown in this paper could also be implemented to find causal relationship between certain google searches and natural events. For example, we could check if there is any correlation between the increase of users searching the words "water", "rain", or "cloud" and the likelihood of a flood happening.

7 ACKNOWLEDGMENTS

This work has been supported by the Slovenian research agency.

8 REFERENCES

[1] M. Michael in S. L. Bressler, „Foundational perspectives on causality in large-scale brain networks," *Physics of Life Reviews,* pp. 107-123, 2015.

[2] G. Dörgő, V. Sebestyén in J. Abonyi, „Evaluating the Interconnectedness of the Sustainable Development Goals Based on the Causality Analysis of Sustainability Indicators," *Sustainability,* 2018.

[3] C. Stefano in S. Sangwon, „Cause-effect analysis for sustainable development policy," *NRC Research Press,* 2017.

[4] https://unstats.un.org/sdgs/indicators/database/.

[5] https://datacatalog.worldbank.org/search/dataset/0037712/World-Development-Indicators.

[6] B. Corrado in K. Peter, „On the directionality of cortical interactions studied," *Biological Cybernetics,* 1999.

[7] K. Irfan, H. Fujun in P. L. Hoang, „The impact of natural resources, energy consumption, and population growth on environmental quality: Fresh evidence from the United States of America," *Science of The Total Environment,* 2020.

[8] H. Tomáš, J. Svatava and M. Bedřich, "Sustainable Development Goals: A need for relevant indicators," *Ecological Indicators,* pp. 565-573, 2016.

Active Learning for Automated Visual Inspection of Manufactured Products

Elena Trajkova*
University of Ljubljana, Faculty of Electrical Engineering
Ljubljana, Slovenia
trajkova.elena.00@gmail.com

Jože M. Rožanec*
Jožef Stefan International Postgraduate School
Ljubljana, Slovenia
joze.rozanec@ijs.si

Paulien Dam
Philips Consumer Lifestyle BV
Drachten, The Netherlands
paulien.dam@philips.com

Blaž Fortuna
Qlector d.o.o.
Ljubljana, Slovenia
blaz.fortuna@qlector.com

Dunja Mladenić
Jožef Stefan Institute
Ljubljana, Slovenia
dunja.mladenic@ijs.si

ABSTRACT

Quality control is a key activity performed by manufacturing enterprises to ensure products meet quality standards and avoid potential damage to the brand's reputation. The decreased cost of sensors and connectivity enabled an increasing digitalization of manufacturing. In addition, artificial intelligence enables higher degrees of automation, reducing overall costs and time required for defect inspection. In this research, we compare three active learning approaches and five machine learning algorithms applied to visual defect inspection with real-world data provided by *Philips Consumer Lifestyle BV*. Our results show that active learning reduces the data labeling effort without detriment to the models' performance.

CCS CONCEPTS

• **Information systems → Data mining**; • **Computing methodologies → Computer vision problems**; • **Applied computing**;

KEYWORDS

Smart Manufacturing, Machine Learning, Automated Visual Inspection, Defect Detection

ACM Reference Format:
Elena Trajkova, Jože M. Rožanec, Paulien Dam, Blaž Fortuna, and Dunja Mladenić. 2021. Active Learning for Automated Visual Inspection of Manufactured Products. In *Ljubljana '21: Slovenian KDD Conference on Data Mining and Data Warehouses, October, 2021, Ljubljana, Slovenia.* ACM, New York, NY, USA, 4 pages.

1 INTRODUCTION

Quality control is one of the critical activities that must be performed by manufacturing enterprises [27, 28]. The main purpose of such activity is to detect product defects meeting quality standards, avoid rework, supply chain disruptions, and avoid potential damage to the brand's reputation [3, 27]. Along with the information regarding defective products, it provides insights into when and where such defects occur, which can be used to further dig into the root causes of such defects and mitigation actions to improve the quality of manufacturing products and processes.

The decreased cost of sensors and connectivity enabled an increasing digitalization of manufacturing [3], which along with the adoption of Artificial Intelligence (AI) [12], represents an opportunity towards enhancing the defect detection in industrial settings [5]. While the quality of the manual inspection has low scalability (requires time to train an inspector, the employees can work a limited amount of time and are subject to fatigue, and the inspection itself is slow), its quality can be affected by the operator-to-operator inconsistency, and it depends on the complexity of the task, the employees (e.g., their intelligence, experience, well-being), the environment (e.g., noise and temperature), the management support and communication [23]; none of these factors affect the outcome of automated quality inspection. Machine learning has been successfully applied to defect detection in a wide range of scenarios [1, 9, 11, 15, 21].

An annotated dataset must be acquired to implement machine learning models for defect detection successfully. The increasing number of sensors provides large amounts of data. As the manufacturing process quality increases, the data obtained from the sensors is expected to be highly imbalanced: most of the data instances will correspond to non-defective products, and a small proportion of them will correspond to different kinds of defects. Annotating all the data is prone to similar limitations as manual inspection described in the paragraph above. It is thus imperative to provide strategies to select a limited subset of them that are most informative to the defect detection models.

We frame the defect detection problem as a supervised learning problem. Given a large amount of unlabeled data, and based on the premise that only a tiny fraction of the data provides new information to the model and thus has the potential to enhance its performance, we adopt an active learning approach. Active learning is a subfield of machine learning that attempts to identify the most informative unlabeled data instances, for which labels are requested some *oracle* (e.g., a human expert) [24]. This research compares three active learning strategies: pool-based sampling, stream-based sampling, and query by committee.

*Both authors contributed equally to this research.

SiKDD '21, October, 2021, Ljubljana, Slovenia

The main contributions of this research are (i) a comparative study between the five most frequently cited machine learning algorithms for automated defect detection and (ii) three active learning approaches (iii) for a real-world multiclass classification problem. We develop the machine learning models with images provided by the *Philips Consumer Lifestyle BV* corporation. The dataset comprises shaver images divided into three classes, based on the defects related to the printing of the logo of the *Philips Consumer Lifestyle BV* corporation: good shavers, shavers with double printing, and shavers with interrupted printing.

We evaluate the models using the area under the receiver operating characteristic curve (AUC ROC, see [4]). AUC ROC is widely adopted as a classification metric, having many desirable properties such as being threshold independent and invariant to a priori class probabilities. We measure AUC ROC considering prediction scores cut at a threshold of 0.5.

This paper is organized as follows. Section 2 outlines the current state of the art and related works, Section 3 describes the use case, and Section 4 provides a detailed description of the methodology and experiments. Finally, section 5 outlines the results obtained, while Section 6 concludes and describes future work.

2 RELATED WORK

Among the many techniques used for automated defect inspection, we find the automated visual inspection, which refers to image processing techniques for quality control, usually applied in the production line of manufacturing industries [1]. Visual inspection requires extracting features from the images, which are used to train the machine learning model. This procedure is simplified when using deep learning models, enabling end-to-end learning, where a single architecture can perform feature extraction and classification [10, 18], and have shown state-of-the-art performance for image classification [20].

The use of automated visual inspection for defect detection has been applied to multiple manufacturing use cases. [21] manually extracted features (e.g., histograms) from machine component images and compared the performance of the Näive Bayes and C4.5 models. [9] extracted statistical features from the images and compared the performance of Support Vector Machines (SVM), Multilayer Perceptron (MLP), and k-nearest neighbors (kNN) models for visual inspection of microdrill bits in printed circuit board production. [11] used 3D convolutional filters applied on computed tomography images and an SVM classifier for defect detection during metallic powder bed fusion in additive manufacturing. [15] used some heuristics to detect regions of interest on slate slab images, on which they performed feature engineering to later train an SVM model on them. Finally, [1] reported using a custom neural network for feature extraction and an SVM model for classification when inspecting aerospace components.

While the authors cited above worked with fully labeled datasets, a production line continually generates new data, exceeding the labeling capacity. A possible solution to this issue is the use of active learning, where the active learner identifies informative unlabeled instances and requests labels to some *oracle*. Typical scenarios involve (i) membership query synthesis (a synthetic data instance is generated), (ii) stream-based selective sampling (the unlabeled instances are drawn one at a time, and a decision is made whether a label is requested, or the sample is discarded), and (iii) pool-based selective sampling (queries samples from a pool of unlabeled data). Among the frequently used querying strategies, we find (i) uncertainty sampling (select an unlabeled sample with the highest uncertainty, given a certain metric or machine-learning model[17]), or (ii) query-by-committee (retrieve the unlabeled sample with the highest disagreement between a set of forecasting models (*committee*)) [6, 24]. More recently, new scenarios have been proposed leveraging reinforcement learning, where an agent learns to select images based on the similarity relationship between the instances and rewards obtained based on the oracle's feedback [22]. In addition, it has been demonstrated that ensemble-based active learning can effectively counteract class imbalance through new labeled images acquisition [2].

Active learning was successfully applied in the manufacturing domain, but scientific literature remains scarce on this domain [19]. Some use cases include the automatic optical inspection of printed circuit boards[8] and the identification of the local displacement between two layers on a chip in the semi-conductor industry[25].

The use of machine learning automates the defect detection, and active learning enables an *inspection by exception* [5], only querying for labels of the images that the model is most uncertain about. While this considerably reduces the volume of required inspections, it is also essential to consider that it can produce an incomplete ground truth by missing the annotations of defective parts classified as false negatives and not queried by the active learning strategy [7].

3 USE CASE

The use case provided for this research corresponds to visual inspection of shavers produced by *Philips Consumer Lifestyle BV*. The visual quality inspection aims to detect defective printing of a logo on the shavers. This use case focuses on four pad printing machines setup for a range of different products, and different logos. A lot of products are produced every day on these machines, which are manually handled and inspected on their visual quality and removed from further processing if the prints on the products are not classified as good. Operators spend several seconds handling, inspecting, and labeling the products. Given an automated visual quality inspection system would strongly reduce the need to manually inspect and label the images, it could speed up the process for more than 40%. Currently there are two types of defects classified related to the printing quality of the logo on the shaver: double printing, and interrupted printing. Therefore, images are classified into three classes: good printing (class zero), double printing (class one), and interrupted printing (class two). A labeled dataset with a total of 3.518 images was provided to train and test the models.

4 METHODOLOGY

We pose automated defect detection as a multiclass classification problem. We measure the model's performance with the AUC ROC metric, using the "one-vs-rest" heuristic method, which involves splitting the multiclass dataset into multiple binary classification problems. Furthermore, we calculate the metrics for each class and

compute their average, weighted by the number of true instances for each class.

To extract features from the images, we make use of the ResNet-18 model [13], extracting embeddings from the Average Pooling layer. Since the embedding results in 512 features, which could cause overfitting, we use the mutual information to evaluate the most relevant ones and select the *top K* features, with $K = \sqrt{N}$, where N is the number of data instances in the train set, as suggested in [14].

To evaluate the models' performance across different active learning strategies, we apply a stratified k-fold cross validation [29], using one fold for testing, one fold as a pool of unlabeled data for active learning, and the rest from training the model. We adopt *k=10* based on recommendations by [16], and query all available unlabeled instances to evaluate the active learning approaches. We compare three active learning scenarios: drawing queries through (i) stream-based classifier uncertainty sampling accepting instances with an uncertainty threshold above the 75^{th} percentile of observed instances, (ii) pool-based sampling selecting the instances a given model is most uncertain about, and pool-based sampling considering a query-by-committee strategy, where the committee is created with models trained with the five algorithms we consider in this research: Gaussian Näive Bayes, CART (*Classification and Regression Trees*, similar to C4.5, but it does not compute rule sets), Linear SVM, MLP, and kNN. Comparing deep learning models remains a subject of future work. Finally, we compare the performance of the active learning scenarios computing the average AUC ROC of each fold and assess if the results differences obtained from each model are statistically significant by using the Wilcoxon signed-rank test[26], using a p-value of 0.05.

5 RESULTS AND ANALYSIS

The results obtained from the experiments we ran, and described in Section 4, are presented in Table 1, and Table 2. Table 1 describes the average AUC ROC per each active learning scenario and model for each cross-validation test fold. We observe that the best model across strategies is the MLP, which achieved the best or second-best performance across almost every fold in pool-based and stream-based active learning. Among those two scenarios, the best results were obtained for stream-based active learning. We observed the same across the rest of the models, though the differences were not significant for all but the Näive Bayes models (see Table 2). Query-by-committee displayed a strong performance, showing best results immediately after the MLP. When assessing the statistical significance between the query-by-committee scenario and results obtained from different models with stream-based and pool-based strategies, we observed that differences were significant in all cases, except for the SVM models. SVM models, most widely used in active learning literature related to automated defect inspection, were the third-best models among the tested ones, immediately after the MLPs in stream-based and pool-based active learning and the query-by-committee approach. SVM models did not display significant differences when compared across different active learning scenarios. The worst results were consistently observed for the CART models.

When analyzing the results, we were interested in how the models' performance evolved through time and significant variations between the first and last results observed. To that end, we assessed the statistical significance between the means of the first and last quartiles of the test fold for each active learning scenario. We assessed the statistical significance using the Wilcoxon signed-rank test, with a p-value of 0.05. While such variations existed and were positive in most test folds (the models learned through time), the improvements were not statistically significant in none of the scenarios.

6 CONCLUSION

In this paper, we compared three active learning scenarios (pool-based, stream-based with classifier uncertainty sampling, and query-by-committee) across five machine learning algorithms (Gaussian Näive Bayes, CART, Linear SVM, MLP, and kNN). We found that the best performance was achieved by the MLP model regardless of the active learning strategy. The second-best performance was obtained through the query-by-committee strategy, while the frequently used SVM models ranked third. We found no significant difference between using pool-based or stream-based active learning approaches. Results from the query-by-committee approach were statistically significant in all cases and better than all the models, except for the MLPs. Finally, we found no case where the improvement between the first and last quartile of the test fold in each active learning scenario would be significant. We believe that further investigation is required to determine if a larger pool of unlabeled images would help us achieve such a significant difference. Future work will focus on data augmentation techniques that could help achieve a statistically significant improvement over time when applying active learning techniques.

ACKNOWLEDGMENTS

This work was supported by the Slovenian Research Agency and the European Union's Horizon 2020 program project STAR under grant agreement number H2020-956573. The authors acknowledge the valuable input and help of Jelle Keizer and Yvo van Vegten from *Philips Consumer Lifestyle BV*.

REFERENCES

[1] Carlos Beltrán-González, Matteo Bustreo, and Alessio Del Bue. 2020. External and internal quality inspection of aerospace components. In *2020 IEEE 7th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*. IEEE, 351–355.

[2] William H Beluch, Tim Genewein, Andreas Nürnberger, and Jan M Köhler. 2018. The power of ensembles for active learning in image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 9368–9377.

[3] Tajeddine Benbarrad, Marouane Salhaoui, Soukaina Bakhat Kenitar, and Mounir Arioua. 2021. Intelligent machine vision model for defective product inspection based on machine learning. *Journal of Sensor and Actuator Networks* 10, 1 (2021), 7.

[4] Andrew P. Bradley. 1997. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition* 30, 7 (1997), 1145 – 1159. https://doi.org/10.1016/S0031-3203(96)00142-2

[5] Amal Chouchene, Adriana Carvalho, Tânia M Lima, Fernando Charrua-Santos, Gerardo J Osório, and Walid Barhoumi. 2020. Artificial intelligence for product quality inspection toward smart industries: quality control of vehicle non-conformities. In *2020 9th international conference on industrial technology and management (ICITM)*. IEEE, 127–131.

[6] David Cohn, Les Atlas, and Richard Ladner. 1994. Improving generalization with active learning. *Machine learning* 15, 2 (1994), 201–221.

[7] Antoine Cordier, Deepan Das, and Pierre Gutierrez. 2021. Active learning using weakly supervised signals for quality inspection. *arXiv preprint arXiv:2104.02973*

Active Learning scenario	Model	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10
stream-based	CART	0,8168	0,7828	0,7810	0,7694	0,8196	0,7805	0,7843	0,7970	0,8409	0,7940
	kNN	0,9289	0,9121	0,9174	0,8686	0,9024	0,9000	0,9051	0,8960	0,9282	0,9082
	MLP	**0,9900**	**0,9928**	**0,9846**	**0,9563**	**0,9804**	**0,9807**	**0,9710**	**0,9729**	0,9793	**0,9845**
	Näive Bayes	0,8818	0,8668	0,8819	0,8686	0,8829	0,8899	0,8650	0,8877	0,8864	0,9098
	SVM	0,9752	0,9828	0,9725	*0,9530*	0,9816	0,9720	0,9570	0,9412	0,9824	0,9712
pool-based	CART	0,7584	0,7904	0,7543	0,7468	0,8441	0,7730	0,8044	0,7701	0,7850	0,7412
	kNN	0,9189	0,9149	0,9161	0,8581	0,9055	0,9036	0,8961	0,8910	0,9224	0,9056
	MLP	*0,9892*	*0,9921*	*0,9845*	**0,9563**	0,9790	**0,9803**	**0,9702**	**0,9723**	0,9806	*0,9840*
	Näive Bayes	0,8800	0,8654	0,8809	0,8677	0,8813	0,8895	0,8637	0,8873	0,8850	0,9090
	SVM	0,9752	0,9819	0,9726	0,9518	*0,9806*	0,9712	0,9562	0,9412	*0,9823*	0,9722
query-by-committee		0,9774	0,9824	0,9714	0,9500	0,9723	*0,9726*	*0,9597*	*0,9571*	**0,9830**	0,9734

Table 1: AUC ROC values were obtained across the ten cross-validation folds. Best results are bolded, second-best results are highlighted in italics.

Model	Active Learning scenarios		
	stream-based vs. pool-based	stream-based vs. query-by-committee	pool-based vs. query-by-committee
CART	0,0840	**0,0020**	**0,0020**
kNN	0,1309	**0,0020**	**0,0020**
MLP	0,0856	**0,0039**	**0,0039**
Näive Bayes	**0,0020**	**0,0020**	**0,0020**
SVM	0,1824	0,4316	0,6250

Table 2: p-values obtained for the Wilcoxon signed-rank test when comparing the average of AUC ROC results across ten cross-validation folds.

(2021).
[8] Wenting Dai, Abdul Mujeeb, Marius Erdt, and Alexei Sourin. 2018. Towards automatic optical inspection of soldering defects. In *2018 International Conference on Cyberworlds (CW)*. IEEE, 375–382.
[9] Guifang Duan, Hongcui Wang, Zhenyu Liu, and Yen-Wei Chen. 2012. A machine learning-based framework for automatic visual inspection of microdrill bits in PCB production. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42, 6 (2012), 1679–1689.
[10] Tobias Glasmachers. 2017. Limits of end-to-end learning. In *Asian Conference on Machine Learning*. PMLR, 17–32.
[11] Christian Gobert, Edward W Reutzel, Jan Petrich, Abdalla R Nassar, and Shashi Phoha. 2018. Application of supervised machine learning for defect detection during metallic powder bed fusion additive manufacturing using high resolution imaging. *Additive Manufacturing* 21 (2018), 517–528.
[12] Irlán Grangel-González. 2019. *A knowledge graph based integration approach for industry 4.0*. Ph.D. Dissertation. Universitäts-und Landesbibliothek Bonn.
[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.
[14] Jianping Hua, Zixiang Xiong, James Lowey, Edward Suh, and Edward R Dougherty. 2005. Optimal number of features as a function of sample size for various classification rules. *Bioinformatics* 21, 8 (2005), 1509–1515.
[15] Carla Iglesias, Javier Martínez, and Javier Taboada. 2018. Automated vision system for quality inspection of slate slabs. *Computers in Industry* 99 (2018), 119–129.
[16] Max Kuhn, Kjell Johnson, et al. 2013. *Applied predictive modeling*. Vol. 26. Springer.
[17] David D Lewis and Jason Catlett. 1994. Heterogeneous uncertainty sampling for supervised learning. In *Machine learning proceedings 1994*. Elsevier, 148–156.
[18] Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3431–3440.
[19] Lingbin Meng, Brandon McWilliams, William Jarosinski, Hye-Yeong Park, Yeon-Gil Jung, Jehyun Lee, and Jing Zhang. 2020. Machine learning in additive manufacturing: A review. *Jom* 72, 6 (2020), 2363–2377.
[20] Samira Pouyanfar, Saad Sadiq, Yilin Yan, Haiman Tian, Yudong Tao, Maria Presa Reyes, Mei-Ling Shyu, Shu-Ching Chen, and Sundaraja S Iyengar. 2018. A survey on deep learning: Algorithms, techniques, and applications. *ACM Computing Surveys (CSUR)* 51, 5 (2018), 1–36.
[21] S Ravikumar, KI Ramachandran, and V Sugumaran. 2011. Machine learning approach for automated visual inspection of machine components. *Expert systems with applications* 38, 4 (2011), 3260–3266.
[22] Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Xiaojiang Chen, and Xin Wang. 2020. A survey of deep active learning. *arXiv preprint arXiv:2009.00236* (2020).
[23] Judi E See. 2012. Visual inspection: a review of the literature. *Sandia Report SAND2012-8590, Sandia National Laboratories, Albuquerque, New Mexico* (2012).
[24] Burr Settles. 2009. Active learning literature survey. (2009).
[25] Karin van Garderen. 2018. Active Learning for Overlay Prediction in Semiconductor Manufacturing. (2018).
[26] Frank Wilcoxon. 1992. Individual comparisons by ranking methods. In *Breakthroughs in statistics*. Springer, 196–202.
[27] Thorsten Wuest, Christopher Irgens, and Klaus-Dieter Thoben. 2014. An approach to monitoring quality in manufacturing using supervised machine learning on product state data. *Journal of Intelligent Manufacturing* 25, 5 (2014), 1167–1180.
[28] Jing Yang, Shaobo Li, Zheng Wang, Hao Dong, Jun Wang, and Shihao Tang. 2020. Using deep learning to detect defects in manufacturing: a comprehensive survey and current challenges. *Materials* 13, 24 (2020), 5755.
[29] Xinchuan Zeng and Tony R Martinez. 2000. Distribution-balanced stratified cross-validation for accuracy estimation. *Journal of Experimental & Theoretical Artificial Intelligence* 12, 1 (2000), 1–12.

Learning to Automatically Identify Home Appliances

Dan Lorbek Ivančič[1], Blaž Bertalanič[1,2], Gregor Cerar[1], Carolina Fortuna[1]

[1]*Jozef Stefan Institute, Ljubljana, Slovenia*
[2]*Faculty of Electrical Engineering, University of Ljubljana, Slovenia*
E-mail: dl0586@student.uni-lj.si

Abstract. Appliance load monitoring (ALM) is a technique that enables increasing the efficiency of domestic energy usage by obtaining appliance specific power consumption profiles. While machine learning have been shown to be suitable for ALM, the work on analyzing design trade-offs during the feature and model selection steps of the ML model development is limited. In this paper we show that 1) statistical features capturing the shape of the time series, yield superior performance by up to 20 percentage points and 2) our best deep neural network-based model slightly outperforms our best gradient descent boosted decision trees by 2 percentage points at the expense of increased training time.

1 Introduction

Household energy consumption accounts for a large proportion of the world's total energy consumption. The first studies, conducted as early as the 1970s, showed that as much as 25% of national energy was consumed by our domestic appliances alone. This figure rose to 30% in 2001 [1] and continues to increase with an exponential rate. Some researchers even predict that these numbers will double by 2030 [2].

In support of rationalizing consumption, appliance load monitoring (ALM) has been introduced. It aims to help solve domestic energy usage related issues by obtaining appliance specific power consumption profiles. Such data can help devise load scheduling strategies for optimal energy utilization [2]. Additionally, data about appliance usage can provide useful insight into daily activities of residents which can be useful for long-distance monitoring of elderly people who prefer to stay at home rather than going to retirement homes [2]. Other applications include theft detection, building safety monitoring, etc.

The two different ways of realizing ALM are intrusive load monitoring (ILM) and non-intrusive load monitoring (NILM). While ILM is known to be more accurate, it requires multiple sensors throughout the entire building to be installed which incurs extra hardware cost and installation complexity. NILM, however, is a cost-effective, easy to maintain process for analyzing changes in the voltage [3] and current going into a building without having to install any additional sensors on different household devices, since it operates using only data obtained from the single main smart meter in a building. The obtained data is then disaggregated and each individual appliance and its energy consumption are detected.

One promising approach to ILM for automatic identification of home appliances is the use of machine learning (ML). For instance, in [4] they used ML to find patterns in the data and extract useful information such as type of load, electricity consumption detail and the running conditions of appliances [4]. More recently, [5] focused on the study of design trade-offs during the feature and model selection steps of the development of the ML-based classifier for ILM. In their study they considered various statistical summaries for feature engineering and classical machine learning techniques for model selection. We complement the work in [5] by extending the feature set with additional shape capturing values and considering deep learning (DNN) and gradient boosted trees (XGBoost) as promising modelling techniques. The contributions of this paper are as follows:

- We explore a variety of different statistical features and show the ones capturing the shape of the time series, such as *longest strike above mean, longest strike below mean, absolute energy and kurtosis* yield superior performance by up to 20 percentage points.
- We show that our best DNN based model slightly outperforms our best XGBoost by 2 percentage points at the expense of increased training time. We also show that our models outperform the results from [5] by 5 percentage points.

The paper is organized as follows. Section 2 summarizes related work, Section 3 formulates the problem and provides methodological details, Section 4 focuses on the study of feature selection trade-offs, while Section 5 discusses model selection. Concluding remarks are drawn in Section 6.

2 Related Work

Existing work that uses machine learning for ALM, such as in [6] investigates the performance of deep learning neural networks on NILM classification tasks and builds a model that is able to accurately detect activations of common electrical appliances using data from the smart meter. More complex DNNs for NILM classification tasks are presented by the authors in [3], where they introduce

a Long Short-Term Memory Recurrent Neural Network (LSTM-RNN) based model and show that it outperforms the considered baselines. In [7] they approach a similar problem by proposing a convolutional neural network based model that allows simultaneous detection and classification of events without having to perform double processing. In [8] authors train a temporal convolutional neural network to automatically extract high-level load signatures for individual appliances while in [9] a feature extraction method is presented using multiple parallel convolutional layers as well as an LSTM recurrent neural network based model is proposed.

3 Problem formulation

Our goal was to design a classifier that when given an input time series T, it is able to accurately map this data to the appropriate class C, as shown in equation 1.

$$C = \Phi(T) \tag{1}$$

where Φ represents the mapping function from time series to target classes and C is a set of these classes, where each class corresponds to one of the following household appliances: computer monitor, laptop computer, television, washer dryer, microwave, boiler, toaster, kettle and fridge. The appliances and measured data illustrated in Figure 1 available in the public UK-Dale dataset are used. The UK DALE (Domestic Appliance-level Electricity) contains the power demand from 5 different houses in the United Kingdom. The dataset was build at a sample-rate of 16 Hz for the whole-house and 0.1667 Hz for each individual appliance. Data is spread into 1 hour long segments, each dataset sample contains a time series with 600 datapoints as depicted in Figure 1.

For realizing Φ, we perform first a feature selection task followed by a model selection one. For selecting the best feature set, we perform feature selection in Section 4. For model selection, we go beyond the work in [5] and consider deep learning architectures enabled by Tensorflow and advanced decision trees that use on optimized distributed gradient boosting technique available in the XGBoost open source library as detailed in Section 5.

4 Feature selection

As can be seen in Figure 1, the time-series corresponding to each device has unique shape and patterns, therefore an intuitive approach to feature selection is to extract statistical properties of the time series that would capture the unique properties of the signals. For instance, a summary such as the peak-to-peak value is able to capture the difference between the maximum and minimum value in a time series signal while one such as skewness is able to describe the asymmetry in the distribution of datapoints in a particular sample. A good combination of such feature would be able to inform the model with relevant information about the power consumption of each appliance, making it easier to find patterns in the data and perform classification task more accurately. Recently, standard tools for computing a large range of such summaries

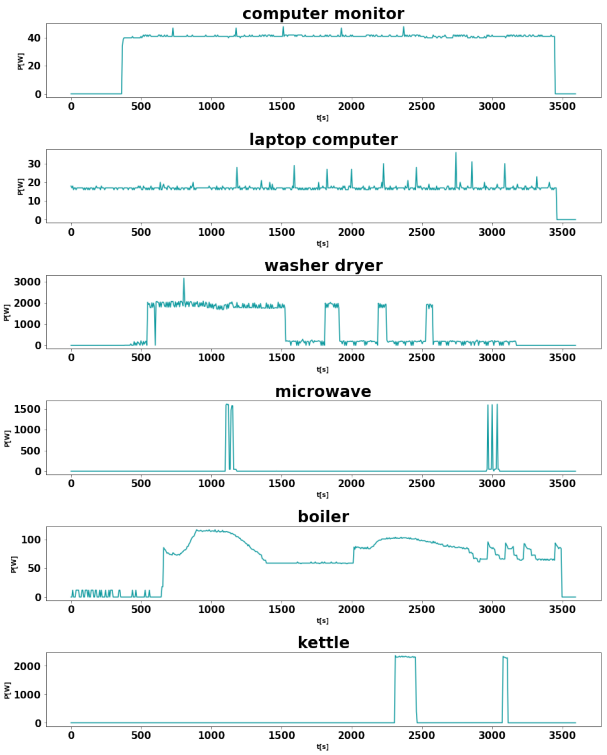

Figure 1: Selected appliances, showing power in relation to time over a 1 hour interval.

are provided by dedicated time series feature engineering tools such as tsfresh [1].

Following an extensive evaluation of combinations of time-series, we report the results for a representative selection of three feature sets as follows:

FeatureSet1 - This feature set consists of the raw time series, containing 2517 time series samples, each with 600 datapoints. It is used as a baseline to see the performance achieves with the available data.

FeatureSet2 - This feature set consists of: $mean$ $value, maximum,$ $minimum, standard\ deviation,$ $variance, peak-to-peak, count\ above\ mean, count$ $below\ mean, mean\ change,$ $absolute\ mean\ change,$ $absolute\ energy$. The count above and below mean counts the numbers of values in each sample that are higher or lower than the mean value of that same sample and helps quantifying the width of a pulse such as the ones for the toaster and microwave from Figure 1. The mean absolute change gives the mean over the absolute differences between subsequent time series values. The absolute energy represents the sum of squared values, calculated using formula shown in equation 2 and provides the information on whether a specific appliance has large consumption profile or not.

$$E = \sum_{i=0}^{n-1} x_i^2 \tag{2}$$

FeatureSet3 - After taking a deeper look into the features from FeatureSet2, we noticed that minimum is re-

[1] $https://tsfresh.readthedocs.io/en/latest/text/list\_of\_features.html$

dundant as it is usually zero in every sample and peak-to-peak is in most cases equal to maximum value due to the lowest value mostly being zero. This feature set consists of: $maximum$, $standard\ deviation$, $mean\ absolute\ change$, $mean\ change$, $longest\ strike\ above\ mean$, $longest\ strike\ below\ mean$, $absolute\ energy$, $kurtosis$, $number\ of\ peaks\ in\ each\ signal$. The longest strike above and below mean returns the length of the the longest consecutive subsequence that is higher or lower than the mean value of that specific sample. The kurtosis is another metric of describing the probability distribution and measures how heavily the tails of a distribution differ from the tails of a normal distribution.

Table 1: Feature comparison using the best models.

Model	Feature set	Precision	Recall	f1
DNN3	FeatureSet1	0.638	0.595	0.573
XGB3	FeatureSet1	0.799	0.769	0.779
DNN3	FeatureSet2	0.918	0.885	0.889
XGB3	FeatureSet2	0.869	0.864	0.867
DNN3	FeatureSet3	**0.931**	**0.898**	**0.902**
XGB3	FeatureSet3	**0.888**	**0.889**	**0.889**
DNN3	best[5]	0.893	0.887	0.888
XGB3	best[5]	0.861	0.860	0.861
SVM[5]	best[5]	0.851	0.835	0.834

4.1 Results

The results of the feature selection process are listed in Table 1 for the two techniques considered in this paper. As can be seen from the second column of the table entitles instances, the dataset is balanced. From columns 3-5 it can be seen that for the baseline FeatureSet1, the f1 score is 0.57 for the CNN and 0.77 for XGB. By using features that better capture the shape of the time series such as in the case of FeatureSet2, an improvement of up to 20% can be seen as follows: the f1 of the CNN model increasing to 0.89, the precision 0.92 and recall to 0.88. The XGBoost model also performed better with an f1 of 0.87, precision of 0.87 and recall of 0.86. Finally, it can be seen from the table that FeatureSet3 performs the best with the f1 of 0.90, precision of 0.93 and recall of 0.90 for the CNN model and f1 of 0.89, precision of 0.89 and recall of 0.89 for the XGB model. FeatureSet3 performed better than FeatureSet2 because its features had much less correlation between each other as well as all of the redundant features from FeatureSet2 were removed. For FeatureSet3, a variety of different feature orderings were also tested but the results remained more within 1% accuracy variance.

To gain insights into the per class performance of FeatureSet3 with the two techniques, we present per device f1 score breakdown in Table 2. It can be seen that computer monitor, microwave and kettle are classified worst by all three models, as their similar consumption profiles make it difficult for the models to distinguish between them. Nevertheless, the CNN classifies all three the best due to its superior pattern recognition ability.

Table 2: Per class performance, FeatureSet3 vs best [5]

Class	Inst.	CNN f1	XGB f1	[5] f1
monitor	300	0.827	0.833	0.780
laptop	276	0.983	0.932	0.838
television	300	0.992	0.976	0.941
washer/dryer	226	0.941	0.912	0.804
microwave	300	0.688	0.620	0.687
boiler	300	1.000	0.968	0.940
toaster	215	0.949	0.940	0.806
kettle	300	0.756	0.722	0.739
fridge	300	1.000	0.983	0.970

5 Model selection

For analyzing the performance of DNN and XGBoost for our problem we conducted extensive performance evaluations. We started by developing a deep learning sequential model, which at first consisted of three dense layers, each with an arbitrarily chosen number of neurons. By trying different combinations of hyperparameters such as number of neurons, loss functions, optimizers, batch size, number of epochs, number of layers and learning rate, we came closer to finding the best suited model for our problem. For optimizing certain hyperparameters we took advantage of the automatic hyperparameter optimization framework Optuna [2]. We then applied similar optimization techniques on the XGB model, although it's default parameter configuration already gave good results. All the experiments were ran on Google Colab using an instance with Nvidia Tesla K80 GPU and 12.69 GB of RAM.

In this section we present and analyze three representative models from each class, DNN and XGboost respectively.

5.1 Deep neural network

DNN1 - This model consisted of three fully connected dense layers. The first two had 32 neurons each as well as ReLU (rectified linear unit) activation function, while the output layer had nine neurons, each corresponding to one of the nine possible appliances and Softmax activation function.

DNN2 - For this model we took the DNN1 model and added an additional dense layer with 64 neurons as well as changed the activation function to linear in the penultimate layer. With this additional complexity we expected to see better results.

DNN3 - For this model we introduced two 1D convolution layers, first with 128 filters and second with 64. Then we used a flatten layer to reduce the dimensionality of the output space, and make the data compatible with the following dense layer, followed by another (output) dense layer.

[2] $https://optuna.org$

5.2 XGBoost

XGB1 - This is the model with standard configuration, i.e. maximum depth of 3, 100 estimators and learning rate of 0.1.

XGB2 - In this model we increased the maximum depth to 4 as well as first reduced learning rate by 50% (to 0.05) and then increased the number of estimators by 50% (to 200). Doing this gave slightly better results.

XGB3 - For this model we decreased the maximum depth to 2, increased number of estimators to 500 and learning rate to 0.25.

Table 3: Model performance on FeatureSet3.

Model	Precision	Recall	f1	Comp. time
DNN1	0.866	0.851	0.846	10.972s
DNN2	0.900	0.887	0.889	21.026s
DNN3	**0.931**	**0.898**	**0.902**	21.124s
XGB1	0.876	0.863	0.864	1.126s
XGB2	0.884	0.881	0.882	2.518s
XGB3	**0.888**	**0.889**	**0.889**	3.225s
SVM [5]	0.878	0.852	0.852	0.301s

5.3 Results

5.3.1 Classification performance

The classification performance of the models is provided in Table 3. It can be seen that the best performing models are DNN3 with an f1 score of 0.90 and XGB3 with an f1 of 0.88. However, the computation time of XGB3 is only 3.23s while for DNN3 it is 21.12s. The XGB classifier using classical machine learning performed only about 1 percentage point worse than the CNN model, while at the same time being much less complex and able to complete the entire training process about 18 seconds faster than the CNN. In addition, the XGB model is much easier to optimize since it has no hidden layers and a pre-arranged hyperparameter configuration that usually requires no further optimization at all. From the last line of the table it can be seen that the SVM-based model from [5] performs 5 percentage points less than DNN3 on FeatureSet3.

5.3.2 Computation time

The superior performance of the DNN model comes at a cost of increased algorithm complexity and hence longer computation time. As depicted in Table 3 the first DNN model took 10.97 seconds to complete the training process and the best (most complex one) took 21.12 seconds. XGBoost, on the other hand, was much faster with XGB1 taking only 1.12 seconds. The added depth for the XGB2 caused a slight increase in computation time to 2.52 seconds, which further increased to 3.23 seconds due to the high number of estimators used in XGB3. Finally, the state of the art was the fastest to complete the training process taking only 0.3 seconds but scored the worst in terms of performance.

6 Conclusions

In this paper we investigated the design trade-offs during the feature and model selection steps of the development of the ML-based classifier for ILM. After formulating our problem, we first show that by extracting various statistical features from raw time series data and then training our models with these features, we were able to improve f1 score by up to 20 percentage points.

Second, we propose two different ML techniques and our process of developing the proposed models using these. We show that optimizing hyperparameters to better suit our specific problem can improve their respective performance by around 4 percentage points. However, choosing the right features that better capture the shape of the data has a much greater impact on the end results than optimizing the models. We also show that classical machine learning model does not perform significantly worse than the deep neural network based one, while at the same time being less computationally expensive.

References

[1] L. Shorrock, J. Utley *et al.*, *Domestic energy fact file 2003*. Citeseer, 2003.

[2] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, "Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey," *Sensors*, vol. 12, no. 12, pp. 16 838–16 866, 2012.

[3] J. Kim, T.-T.-H. Le, and H. Kim, "Nonintrusive Load Monitoring Based on Advanced Deep Learning and Novel Signature," *Computational Intelligence and Neuroscience*, vol. 2017, p. e4216281, Oct. 2017, publisher: Hindawi. [Online]. Available: https://www.hindawi.com/journals/cin/2017/4216281/

[4] E. Aladesanmi and K. Folly, "Overview of non-intrusive load monitoring and identification techniques," *IFAC-PapersOnLine*, vol. 48, no. 30, pp. 415–420, 2015, 9th IFAC Symposium on Control of Power and Energy Systems CPES 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896315030566

[5] L. Ogrizek, B. Bertalanic, G. Cerar, M. Meza, and C. Fortuna, "Designing a machine learning based non-intrusive load monitoring classifier," in *2021 IEEE ERK*, 2021, pp. 1–4.

[6] M. Devlin and B. P. Hayes, "Non-intrusive load monitoring using electricity smart meter data: A deep learning approach," in *2019 IEEE Power Energy Society General Meeting (PESGM)*, 2019, pp. 1–5.

[7] F. Ciancetta, G. Bucci, E. Fiorucci, S. Mari, and A. Fioravanti, "A new convolutional neural network-based system for nilm applications," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–12, 2021.

[8] Y. Yang, J. Zhong, W. Li, T. A. Gulliver, and S. Li, "Semisupervised multilabel deep learning based nonintrusive load monitoring in smart grids," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 11, pp. 6892–6902, 2020.

[9] W. He and Y. Chai, "An empirical study on energy disaggregation via deep learning," *Advances in Intelligent Systems Research*, vol. 133, pp. 338–342, 2016.

Indeks avtorjev / Author index