

Zbornik 23. mednarodne multikonference

INFORMACIJSKA DRUŽBA

Zvezek A

Proceedings of the 23rd International Multiconference

INFORMATION SOCIETY

Volume A

<http://is.ijs.si>
**IS
20
20**

Slovenska konferenca o
umetni inteligenci

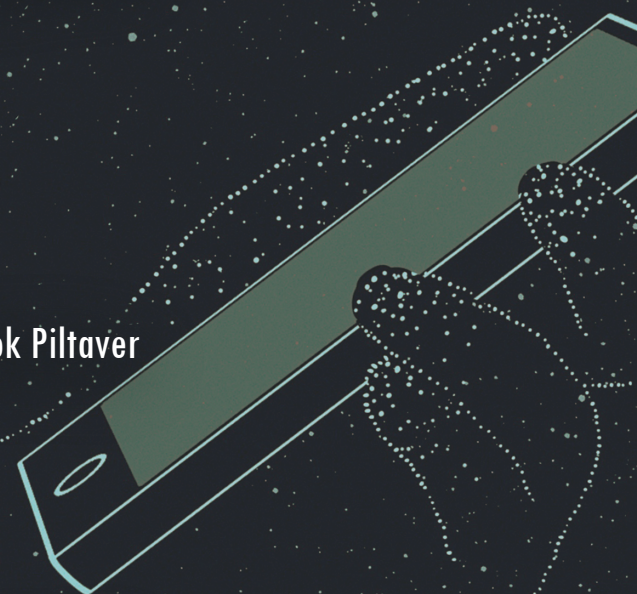
Slovenian Conference on
Artificial Intelligence

Uredili / Edited by

Mitja Luštrek, Matjaž Gams, Rok Piltaver

6.—7. oktober 2020 / 6—7 October 2020

Ljubljana, Slovenia



Zbornik 23. mednarodne multikonference
INFORMACIJSKA DRUŽBA – IS 2020
Zvezek A

Proceedings of the 23rd International Multiconference
INFORMATION SOCIETY – IS 2020
Volume A

Slovenska konferenca o umetni inteligenci
Slovenian Conference on Artificial Intelligence

Uredili / Edited by

Mitja Luštrek, Matjaž Gams, Rok Piltaver

<http://is.ijs.si>

6. – 7. oktober 2020 / 6 - 7 October 2020
Ljubljana, Slovenia

Uredniki:

Mitja Luštrek
Odsek za inteligentne sisteme
Institut »Jožef Stefan«, Ljubljana

Matjaž Gams
Odsek za inteligentne sisteme
Institut »Jožef Stefan«, Ljubljana

Rok Piltaver
Celtra, d. o. o. in
Odsek za inteligentne sisteme
Institut »Jožef Stefan«, Ljubljana

Založnik: Institut »Jožef Stefan«, Ljubljana
Priprava zbornika: Mitja Lasič, Vesna Lasič, Lana Zemljak
Oblikovanje naslovnice: Vesna Lasič

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

Ljubljana, oktober 2020

Informacijska družba
ISSN 2630-371X

Kataložni zapis o publikaciji (CIP) pripravili v Narodni in univerzitetni knjižnici v Ljubljani COBISS.SI-ID=33223427 ISBN 978-961-264-202-0 (epub) ISBN 978-961-264-203-7 (pdf)

PREDGOVOR MULTIKONFERENCI INFORMACIJSKA DRUŽBA 2020

Triindvajseta multikonferenca Informacijska družba (<http://is.ijs.si>) je doživela polovično zmanjšanje zaradi korone. Zahvala za preživetje gre tistim predsednikom konferenc, ki so se kljub prvi pandemiji modernega sveta pogumno odločili, da bodo izpeljali konferenco na svojem področju.

Korona pa skoraj v ničemer ni omejila neverjetne rasti IKTja, informacijske družbe, umetne inteligence in znanosti nasploh, ampak nasprotno – kar naenkrat je bilo večino aktivnosti potrebno opraviti elektronsko in IKT so dokazale, da je elektronsko marsikdaj celo bolje kot fizično. Po drugi strani pa se je pospešil razpad družbenih vrednot, zaupanje v znanost in razvoj. Celó Flynnov učinek – merjenje IQ na svetovni populaciji – kaže, da ljudje ne postajajo čedalje bolj pametni. Nasprotno - čedalje več ljudi verjame, da je Zemlja ploščata, da bo cepivo za korono škodljivo, ali da je korona škodljiva kot navadna gripa (v resnici je desetkrat bolj). Razkorak med rastočim znanjem in vraževerjem se povečuje.

Letos smo v multikonferenco povezali osem odličnih neodvisnih konferenc. Zajema okoli 160 večinoma spletnih predstavitev, povzetkov in referatov v okviru samostojnih konferenc in delavnic in 300 obiskovalcev. Prireditve bodo 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 44-letno tradicijo odlične znanstvene revije.

Multikonferenco Informacijska družba 2020 sestavljajo naslednje samostojne konference:

- Etika in stroka
- Interakcija človek računalnik v informacijski družbi
- Izkopavanje znanja in podatkovna skladišča
- Kognitivna znanost
- Ljudje in okolje
- Mednarodna konferenca o prenosu tehnologij
- Slovenska konferenca o umetni inteligenci
- Vzgoja in izobraževanje v informacijski družbi

Soorganizatorji in podporniki konference so različne raziskovalne institucije in združenja, med njimi tudi 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.

V 2020 bomo petnajstič podelili 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 prejela prof. dr. Lidija Zadnik Stirn. Priznanje za dosežek leta pripada Programskemu svetu tekmovanja ACM Bober. Podeljujemo tudi nagradi »informacijska limona« in »informacijska jagoda« za najbolj (ne)uspešne poteze v zvezi z informacijsko družbo. Limono je prejela »Neodzivnost pri razvoju elektronskega zdravstvenega kartona«, jagodo pa Laboratorij za bioinformatiko, Fakulteta za računalništvo in informatiko, Univerza v Ljubljani. Čestitke nagrajencem!

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

FOREWORD

INFORMATION SOCIETY 2020

The 23rd Information Society Multiconference (<http://is.ijs.si>) was halved due to COVID-19. 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 pandemics did not decrease the growth of ICT, information society, artificial intelligence and science overall, quite on the contrary – suddenly most of the activities had to be performed by ICT and often it was more efficient than in the old physical way. But COVID-19 did increase downfall of societal norms, trust in science and progress. Even the Flynn effect – measuring IQ all over the world – indicates that an average Earthling is becoming less smart and knowledgeable. Contrary to general belief of scientists, the number of people believing that the Earth is flat is growing. Large number of people are weary of the COVID-19 vaccine and consider the COVID-19 consequences to be similar to that of a common flu dispute empirically observed to be ten times worst.

The Multiconference is running parallel sessions with around 160 presentations of scientific papers at twelve conferences, many round tables, workshops and award ceremonies, and 300 attendees. Selected papers will be published in the Informatica journal with its 44-years tradition of excellent research publishing.

The Information Society 2020 Multiconference consists of the following conferences:

- Cognitive Science
- Data Mining and Data Warehouses
- Education in Information Society
- Human-Computer Interaction in Information Society
- International Technology Transfer Conference
- People and Environment
- Professional Ethics
- Slovenian Conference on Artificial Intelligence

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, DKZ 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 thorough reviews.

For the fifteenth year, 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 Prof. Dr. Lidija Zadnik Stirn for her 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 Program Council of the competition ACM Bober. The information lemon goes to the “Unresponsiveness in the development of the electronic health record”, and the information strawberry to the Bioinformatics Laboratory, Faculty of Computer and Information Science, University of Ljubljana. Congratulations!

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

KONFERENČNI ODBORI

CONFERENCE COMMITTEES

International Programme Committee

Vladimir Bajic, South Africa
Heiner Benking, Germany
Se Woo Cheon, South Korea
Howie Firth, UK
Olga Fomichova, Russia
Vladimir Fomichov, Russia
Vesna Hljuz Dobric, Croatia
Alfred Inselberg, Israel
Jay Liebowitz, USA
Huan Liu, Singapore
Henz Martin, Germany
Marcin Paprzycki, USA
Claude Sammut, Australia
Jiri Wiedermann, Czech Republic
Xindong Wu, USA
Yiming Ye, USA
Ning Zhong, USA
Wray Buntine, Australia
Bezalel Gavish, USA
Gal A. Kaminka, Israel
Mike Bain, Australia
Michela Milano, Italy
Derong Liu, Chicago, USA
prof. Toby Walsh, Australia

Organizing Committee

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

Programme Committee

Mojca Cigliarič, 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
Špela Stres
Stanko Strmčnik
Jurij Šilc
Jurij Tasič
Denis Trček
Andrej Ule
Tanja Urbančič
Boštjan Vilfan
Baldomir Zajc
Blaž Zupan
Boris Žemva
Leon Žlajpah

KAZALO / TABLE OF CONTENTS

Slovenska konferenca o umetni inteligenci / Slovenian Conference on Artificial Intelligence	1
PREDGOVOR / FOREWORD.....	3
PROGRAMSKI ODBORI / PROGRAMME COMMITTEES.....	5
Using Mozilla's Deep Speech to Improve Speech Emotion Recognition / Andova Andrejaana, Bromuri Stefano, Luštrek Mitja	7
Towards Automatic Recognition of Similar Chess Motifs / Bizjak Miha, Guid Matej.....	11
Drinking Detection From Videos in a Home Environment / De Masi Carlo M., Luštrek Mitja	15
Semantic Feature Selection for AI-Based Estimation of Operation Durations in Individualized Tool Manufacturing / Dovgan Erik, Filipič Bogdan	19
Generating Alternatives for DEX Models using Bayesian Optimization / Gjoreski Martin, Kuzmanovski Vladimir	23
Detekcija napak na industrijskih izdelkih / Golob David, Petrovčič Janko, Kalabakov Stefan, Kocuvan Primož, Bizjak Jani, Dolanc Gregor, Ravničan Jože, Gams Matjaž, Bohanec Marko	27
Data Protection Impact Assessment - an Integral Component of a Successful Research Project From the GDPR Point of View / Gültekin Várkonyi Gizem, Gradišek Anton.....	32
Deep Transfer Learning for the Detection of Imperfectionson Metallic Surfaces / Kalabakov Stefan, Kocuvan Primož, Bizjak Jani, Gazvoda Samo, Gams Matjaž.....	35
Fall Detection and Remote Monitoring of Elderly People Using a Safety Watch / Kiprijanovska Ivana, Bizjak Jani, Gams Matjaž.....	39
Machine Vision System for Quality Control in Manufacturing Lines / Kiprijanovska Ivana, Bizjak Jani, Gazvoda Samo, Gams Matjaž	43
Abnormal Gait Detection Using Wrist-Worn Inertial Sensors / Kiprijanovska Ivana, Gjoreski Hristijan, Gams Matjaž	47
Avtomatska detekcija obrabe posnemalnih igel / Kocuvan Primož, Bizjak Jani, Kalabakov Stefan, Gams Matjaž	51
Povečevanje enakosti (oskrbe duševnega zdravja) s prepričljivo tehnologijo / Kolenik Tine, Gams Matjaž	55
Analiza glasu kot diagnostičn ametodaza odkrivanje Parkinsonove bolezni / Levstek Andraž, Silan Darja, Vodopija Aljoša.....	59
STRAW Application for Collecting Context Data and Ecological Momentary Assessment / Lukan Junoš, Kutrašnik Marko, Bolliger Larissa, Clays Els, Luštrek Mitja	63
URBANITE H2020 Project Algorithms and Simulation Techniques for Decision-Makers / Machidon Alina, Smerkol Maj, Gams Matjaž	68
Towards End-to-end Text to Speech Synthesis in Macedonian Language / Neceva Marija, Stoilkovska Emilija, Gjoreski Hristijan	72
Improving Mammogram Classification by Generating Artificial Images / Peterka Ana, Bosnić Zoran, Osipov Evgeny.....	76
Mobile Nutrition Monitoring System: Qualitative and Quantitative Monitoring / Reščič Nina, Jordan Marko, De Boer Jasmijn, Bierhoff Ilse, Luštrek Mitja	80
Recognition of Human Activities and Falls by Analyzing the Number of Accelerometers and their Body Location / Shulajkovska Miljana, Gjoreski Hristijan.....	84
Sistem za ocenjevanje esejev na podlag ikoherence in semantične skladnosti / Simončič Žiga, Bosnić Zoran	88
Mental State Estimation of People with PIMD using Physiological Signals / Slapničar Gašper, Dovgan Erik, Valič Jakob, Luštrek Mitja.....	92
Energy-Efficient Eating Detection Using a Wristband / Stankoski Simon, Luštrek Mitja.....	96
Comparison of Methods for Topical Clustering of Online Multi-speaker Discourses / Stropnik Vid, Bosnić Zoran, Osipov Evgeny	100
Machine Learning of Surrogate Models with an Application to Sentinel 5P / Szlupowicz Michał Artur, Brenc Jure, Adams Jennifer, Malina Edward, Džeroski Sašo	104
Deep Multi-label Classification of ChestX-ray Images / Štepec Dejan.....	108
Smart Issue Retrieval Application / Zupančič Jernej, Budna Borut, Mlakar Miha, Smerkol Maj	112
Adaptation of Text to Publication Type / Žontar Luka, Bosnić Zoran	116
Indeks avtorjev / Author index	121

Zbornik 23. mednarodne multikonference
INFORMACIJSKA DRUŽBA – IS 2020
Zvezek A

Proceedings of the 23rd International Multiconference
INFORMATION SOCIETY – IS 2020
Volume A

Slovenska konferenca o umetni inteligenci
Slovenian Conference on Artificial Intelligence

Uredili / Edited by

Mitja Luštrek, Matjaž Gams, Rok Piltaver

<http://is.ijs.si>

6. – 7. oktober 2020 / 5 - 7 October 2020
Ljubljana, Slovenia

PREDGOVOR

Leto 2020 je bilo za informacijsko družbo zelo pomembno: zmanjšanje medosebnih stikov zaradi COVID-19 je pokazalo, da se da s pomočjo informacijskih tehnologij postoriti še precej več, kot smo si do zdaj mislili. S pomočjo telekonferenčnih sistemov smo se sestajali, digitalno smo prenašali in podpisovali dokumente, prek spleta smo lahko naročili domala vse izdelke in storitve ... Čeravno sta umetna inteligenca in informacijska družba vedno tesneje povezani, pa podobno dramatičnega napredka pri umetni inteligenci ni bilo opaziti. Seveda to ne pomeni, da napredka ni bilo – raznotere metode umetne inteligence še naprej postajajo vedno zmogljivejše in predvsem prodirajo v vedno manjše in cenejše naprave: opažamo lahko, da se namenski procesorji za operacije umetnih nevronske mreže vedno pogosteje pojavljajo v pametnih telefonih, pametnih zvočnikih z govornimi asistenti in podobnih napravah.

Umetno inteligenco smo zapregli tudi v spopad s COVID-19. Raziskovalci so jo uporabili za določanje strukture virusa in za iskanje učinkovitih zdravil in cepiv. Skupina ameriških organizacij je razpisala nagrado za najboljše pristope rudarjenja po besedilih, ki bodo iz 19 GB besedil, povezanih z boleznijo, izluščila koristne informacije. Razvitih je bilo več diagnostičnih sistemov za podporo odločanju, ki analizirajo slike pljuč in druge podatke. Precej raziskovalcev se je z metodami umetne inteligence lotilo napovedovanja širjenja bolezni in določanja dejavnikov, ki nanj vplivajo. Tovrstne raziskave se dogajajo tudi v Sloveniji.

K sreči COVID-19 naši konferenci ni storil dosti žalega. Resda se ob pisanju tega uvodnika še ne ve zagotovo, ali bo konferenca potekala na daljavo ali jo bomo uspeli speljati hibridno, kot načrtujemo – da bo del udeležencev prisoten v živo v predavalnici, del pa na daljavo. A verjamemo, da to na kakovost izvedbe ne bo bistveno vplivalo. Z zadovoljstvom pa ugotavljamo, da smo letos dobili največ prispevkov v zadnjih petih letih – v zbornik jih je vključenih kar 28. Tokrat je bolje kot običajno zastopana Fakulteta za računalništvo in informatiko Univerze v Ljubljani, ki ima skupaj z Institutom Jožef Stefan (od koder je – kot vsako leto – največ prispevkov) vodilno vlogo pri raziskavah umetne inteligence v Sloveniji. Nekaj prispevkov je tudi iz tujine in industrije, čeprav bi si zlasti slednjih želeli več. Slovenija namreč izobrazila veliko strokovnjakov s področja umetne inteligence in precej jih najde pot v industrijo, kjer se dogaja marsikaj zanimivega, o čemer vemo premalo. V to smer si bomo zato še bolj prizadevali v prihodnjih letih.

FOREWORD

2020 was an important year for the information society: social distancing due to COVID-19 showed that information technologies allow us to do even more than we previously thought. Teleconferencing systems allowed us to meet virtually, we transferred and signed documents digitally, we ordered every imaginable product and service online ... However, even though artificial intelligence and information society are increasingly interlinked, the progress of artificial intelligence this year was not as significant. This certainly does not mean there was no progress – various artificial-intelligence methods are still steadily improving, and, perhaps even more importantly, becoming available in ever smaller and cheaper devices: dedicated processors accelerating neural-network computations are becoming common in smartphones, smart speakers with conversational assistants and similar devices.

Artificial intelligence also helps fight COVID-19. It was used to determine the structure of the virus and to discover effective drugs and vaccines. A group of USA organizations offered a prize for the best data-mining methods that can extract information from 19 GB of texts related to the disease. Several diagnostic decision support systems were developed, which analyse images of the lungs and other data. Many researchers used artificial intelligence to forecast the spread of the disease and the factors that affect it. Such research is also conducted in Slovenia.

Fortunately, COVID -19 did not much affect our conference. At the time of writing this editorial, it is still not clear whether it will take place remotely, or we will succeed with planned the hybrid approach, where a part of the participants will attend live in a lecture room with the rest connected via teleconference. Either way, we are confident this will not have a major impact on the quality of the conference. We are pleased to report that this year we have the largest number of papers in the last five years – there are 28 in these proceedings. The Faculty of Computer and Information Science is represented better than in previous years, which is quite appropriate considering that – aside from Jožef Stefan Institute (which contributed the largest number of papers, as usual) – it is the leading Slovenian research institution on artificial intelligence. There are also some papers from abroad and from the industry, although we would prefer to see more of these, especially the latter. The number of experts on artificial intelligence in Slovenia is quite large and a significant number find their way to the industry, where many interesting but not widely known developments take place. We aim to improve on this aspect in the following years.

PROGRAMSKI ODBOR / PROGRAMME COMMITTEE

Mitja Luštrek

Matjaž Gams

Rok Piltaver

Marko Bohanec

Tomaž Banovec

Cene Bavec

Jaro Berce

Marko Bonač

Ivan Bratko

Dušan Caf

Bojan Cestnik

Aleš Dobnikar

Bogdan Filipič

Nikola Guid

Borka Jerman Blažič

Tomaž Kalin

Marjan Krisper

Marjan Mernik

Vladislav Rajkovič

Ivo Rozman

Niko Schlamberger

Tomaž Seljak

Miha Smolnikar

Peter Stanovnik

Damjan Strnad

Peter Tancig

Pavle Trdan

Iztok Valenčič

Vasja Vehovar

Martin Žnidaršič

Using Mozilla's DeepSpeech to Improve Speech Emotion Recognition

Andrejaana Andova
Jožef Stefan International
Postgraduate School
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia
andrejaana.andova@ijs.si

Stefano Bromuri
Open University of the Netherlands
Heerlen, Netherlands
Stefano.Bromuri@ou.nl

Mitja Luštrek
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia
mitja.lustrek@ijs.si

ABSTRACT

A lot of effort in detecting emotions in speech has already been made. However, most of the related work was focused on training a model on an emotional speech dataset, and testing the model on the same dataset. A model trained on one dataset seems to provide poor results when tested on another dataset. This means that the models trained on publicly available datasets cannot be used in real-life applications where the speech context is different. Furthermore, collecting large amounts of data to build an efficient speech emotion classifier is not possible in most cases.

Because of this, some researchers tried using transfer learning to improve the performance of a baseline model trained on only one dataset. However, most of the works so far developed methods that transfer information from one emotional speech dataset into another emotional speech dataset.

In this work, we try to transfer parameters from a pre-trained speech-to-text model that is already widely used. Unlike other related work, which uses emotional speech datasets that are usually small, in this method we will try to transfer information from a larger speech dataset which was collected by Mozilla and whose main purpose was to transcribe speech.

We used the first layer from the DeepSpeech model as the basis for building another deep neural network, which we trained on the improvisation utterances from the IEMOCAP dataset.

KEYWORDS

speech emotion recognition, feature transfer, DeepSpeech

1 INTRODUCTION

There are many issues when trying to build a model for speech emotion recognition, but the main problem is the lack of emotional speech data. Collecting a dataset is often a challenging and effortful task, but in speech emotion recognition a few additional problems arise when creating a dataset. One of the main problems is that speech is a context-dependent problem. One could gather a dataset from job interviews and build a precise model that detects emotions in job applicants' speech. However, the same model would probably not work for a phone application that tries to analyze the emotions of its users. Thus, to build a general model for speech emotion recognition, one would need to

gather a dataset composed of speeches used in different contexts, which is a hard task.

Most of the currently available emotional speech datasets are composed of actors performing scenes with different emotions. Finding actors and writing the scenes could be a costly and effortful task and, thus, it is hard to collect large amounts of data in this way. However, the major problem of this type of data is that all of the emotions are acted and may be more exaggerated when compared to real-life emotions [8]. This type of data is probably pretty different when compared to data from real-life applications where emotions are expressed with less intensity. To solve this problem, some researchers tried using transfer learning methods to build a model that is more robust to changes in the data.

Some researchers tried using speeches recorded in real-life scenarios and asked people to listen to these speeches and annotate the emotions they recognize in the speakers' voices. When collecting a dataset in this way one needs to find people that would listen to the whole dataset and annotate the data. The annotators would probably have different abilities to detect the emotions and different perceptions of what each emotion should be like. Because of this, in many cases not all of them will agree on which emotion is present in a sample. Another drawback of this type of data collection is that most of the time people do not experience extreme emotions. Because of this, such datasets will result in almost no emotions – the speech would be mostly neutral.

The main idea behind transfer learning is to use information from a dataset called source dataset to improve the performance of a target dataset. The source and the target datasets may have labeled or unlabeled data, may have the same data distribution or different data distribution, and they can be constructed to solve the same task or they may try to solve different tasks. Depending on this, there are different approaches to transfer learning. They are more thoroughly explained by S. J. Pan et al. [5].

In this work, we decided to follow the usual transfer learning approach, and use a pre-trained speech-to-text model trained on a large nonemotional English dataset collected by Mozilla. This model may not contain any emotional information that would be useful for our task, but we believe it contains information about the speech of the subjects that could be used in speech emotion recognition.

2 RELATED WORK

While research in speech emotion recognition where training and testing are done on one dataset has already been well-studied, using other datasets to make the model more generalized has been in focus only in recent years.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

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

© 2020 Copyright held by the owner/author(s).

Table 1: Emotion distribution in IEMOCAP.

Anger	Happiness	Sadness	Neutral
500	94	467	392

Some researchers tried using unlabeled target data to improve speech emotion recognition models. Thus, Parthasarathy and Busso [6] connected supervised and unsupervised learning to improve the performance of speech emotion recognition on a target dataset. They used a network architecture similar to autoencoders to encode large amounts of unlabeled target data in an unsupervised way by putting the same speech in the input and the output of the network. To force the network to encode the emotional information from the speech, they connected the last encoding layer to another layer that was trying to learn the arousal, valence, and the dominance annotations on the speech in a supervised way. When they compared their method to other state-of-the-art models, it showed improvement in the arousal and the dominance space while in the valence space they got results slightly worse than the state-of-the-art.

Some authors thought about bringing the feature space from the source and the target data closer together. Thus, Song et al., [7] used MMDE optimization and dimension reduction algorithms to bring the feature spaces from the source and the target datasets closer together. After that, they used the shifted feature space from the source dataset to train an SVM model. They used the EmoDB dataset as a source dataset, and a Chinese emotional dataset collected by them as a target dataset. After they trained the SVM model on the source dataset only, they applied the model on the target dataset and showed that the model performed with 59.8% accuracy. These results show improvement when compared to an SVM model trained on the source dataset and tested on the target dataset without any dimension reduction applied, which performs with 29.8% accuracy. However, the best performance was achieved with a model trained and tested on the target dataset, which achieved 85.5% accuracy.

3 DATASET

In this research we used the Interactive emotional dyadic motion capture database (IEMOCAP) [1]. IEMOCAP consists of speech from ten different English-speaking actors (five male and five female), and it is the largest dataset for speech emotion recognition that we found publicly available. It consists of approximately twelve hours of data where actors perform improvisations or scripted scenarios, specifically selected to elicit emotional expressions. Since the actors were not given any specific emotions that they had to act, the database was annotated by multiple annotators into categorical labels, as well as dimensional labels, such as valence, activation, and dominance. The set of emotions the annotators could choose from was anger, happiness, excitement, sadness, frustration, fear, surprise, other, and neutral, but because most of the related work on transfer learning in speech emotion recognition only used anger, happiness, sadness and neutral utterances in their methods, we decided to also just use these emotions in our method.

We noticed that most of the time, the three annotators did not perceive the same emotion and, thus, we decided to eliminate all data where all three annotators did not agree on the detected emotion. This reduced the amount of data significantly. The

distribution of the emotions after the data reduction is given in Table 1.

4 METHODOLOGY

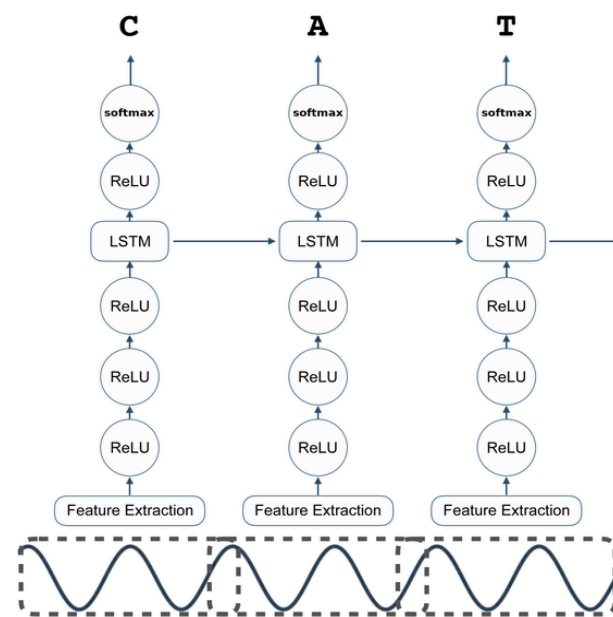
We developed methods that transfer information from a large nonemotional speech dataset into a target emotional speech dataset. Since in most of the related work researchers were extracting information from smaller emotional speech datasets and transferring this information to other emotional speech datasets, this is the first attempt that we know of in which a transfer of information is tried from already well-defined pre-trained speech dataset into a smaller emotional speech dataset, which is the standard approach in most transfer learning applications.

However, to compare if the methods provide any useful improvement, we compare them to a baseline model that was trained and tested on IEMOCAP, and which does not use any kind of information transfer.

4.1 Baseline Model

To build a baseline classifier, we decided to use standard machine learning approaches trained on features extracted using OpenS-MILE [2] as a baseline method. After testing several different machine learning approaches, we saw that Random Forest obtained the best results for most of the target datasets. Because of this, we decided to use a Random Forest classifier with 1000 trees and a maximal depth of 10 as a baseline model.

4.2 DeepSpeech Model

**Figure 1: Architecture of the original DeepSpeech model.**

DeepSpeech is a model that tries to provide transcriptions of a given speech. The model has been trained on the English

Table 2: Classification accuracy obtained from the majority classifier and baseline Random Forest Classifier compared to the DeepSpeech features method.

Model	Majority	Baseline	DeepSpeech features
Dense	34%	67%	58%
LSTM	34%	67%	7%
Dense1+Dense2	34%	67%	26%
Dense1+LSTM2	34%	67%	66%

data from the Mozilla Common Voice dataset [3]. This dataset consists of 1469 hours of speech data that has been recorded by 61521 different voices. The people whose voices were collected belonged to different nationalities (and thus different English accents), and different ages. All of this data is publicly available and can be easily accessed.

The architecture of the DeepSpeech model is presented in Figure 1. Each utterance is a time-series data, where every time-slice is a vector of MFCC audio features [4]. The goal of the network is to convert an input sequence x into a sequence of character probabilities for the transcription y .

The network is composed of five hidden layers. The first three layers are dense layers with ‘ReLU’ as an activation function. The fourth layer is an LSTM layer, the fifth layer is once again a dense layer with ‘ReLU’ activation function. The output layer has a softmax function which outputs character probabilities. In the example in Figure 1 the output of the first frame is the character ‘C’, the second frame outputs the character ‘A’, and the third frame outputs the character ‘T’, resulting with the word ‘CAT’.

4.3 Transfer Learning Using DeepSpeech

We decided to experiment if we could transfer information from the DeepSpeech model that would be useful for the speech emotion recognition task. We used the representation learned by the DeepSpeech network to extract features for the IEMOCAP dataset. We used the output from the first layer in the DeepSpeech model as features for a given frame. We ended up with 2048 features for every 10-millisecond frame. So, if the whole utterance was 3 seconds long, we would receive a matrix with dimensions 1800x2048 after the deep speech feature extraction.

After the features from all the samples in IEMOCAP have been extracted, we trained a deep neural network using them. We simply added the layers from the new deep neural network on top of the first layer from the DeepSpeech model, and trained the new deep neural network from scratch by just using the samples from the IEMOCAP dataset. This way we repurpose the feature representations from the first layer of the DeepSpeech model.

We experimented with several different deep neural network architectures to see which one works best for this problem. In the first architecture, we used a feed-forward network on the extracted features per each frame. We used one hidden dense layer with ‘relu’ activation function and 204 neurons. We connected this layer to a dense layer with softmax activation function which predicted the emotion probabilities for each frame separately. Although in the IEMOCAP dataset there are no labels for each of the frames separately, we use the target label for the whole utterance as target label for each of the frames.

The second model architecture we tried was to use the features from the whole frame as input, and use a LSTM layer to learn the representations from the features. The LSTM layer is activated by

a ‘relu’ function and has 20 hidden states. It is then connected to a dense layer activated by a ‘softmax’ activation function which predicts the label of the whole utterance.

The third network architecture is composed of two parts. In the first part we predict the emotion probabilities for each frame separately and in the second part we use the emotion probabilities predictions from the first layer to predict the emotion probabilities for the whole utterance. The first part of the architecture is the same as in the first network architecture and is trained on one half of the training data. In the second part of this network, we use the predictions from the first part as input to a dense layer with a softmax activation function. The second part of the network is trained on the other half of the training data. In this network architecture, for each sequence of 20 frames we predict one vector of emotions.

The fourth network consists of two separate parts and is presented in Figure 2. The first part takes the output of the DeepSpeech model, and tries to predict the probability for each of the target emotions separately. The first dense layer has a ‘relu’ activation function and outputs 204 features. It is then connected to another dense layer with a softmax activation function that predicts the emotions present in each frame separately. The second part of the network uses the output emotion probabilities from the first part of the layer as an input. The second part of the network consists of one LSTM layer which is trained on the second half of the training data. The LSTM layer is activated by a ‘relu’ function and has 20 hidden states. It is then connected to a dense layer activated by a ‘softmax’ activation function which predicts the label of the whole utterance. This network architecture in a way is a combination from the first and the second network architecture.

5 RESULTS

Since the DeepSpeech model is capable of learning language phases in the speech, we decided to remove all scripted utterances from the IEMOCAP dataset and use just the utterances in which the actors were asked to improvise. To evaluate the neural network architectures we used the leave-one-subject-out cross validation.

In Table 2 we present the results obtained from each of the deep neural network architectures that we tried as well as the accuracy of the baseline model and the majority classifier. In the results we can see that the LSTM network architecture that we tried performs quite poor, with classification accuracy of only 7%. The most probable explanation for this is that this architecture is quite complex since it has 2048 features for each frame, and it tries to train an LSTM model on all of these features. To train a model with this amount of parameters, we would need much more samples than the IEMOCAP improvisations.

The architecture that provides the best results is the one that uses a FFN to predict the features in each frame, and then uses a

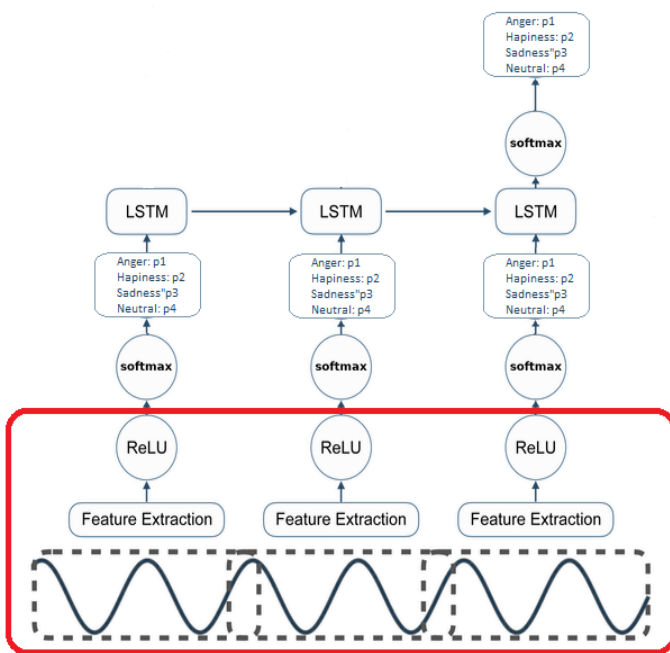


Figure 2: Architecture of the original DeepSpeech model.

LSTM network to predict the final emotion predictions for the whole utterance. We further experimented with this network architecture to see how much the length of the frames changes the performance of the model. The results are presented in Figure 3. In this figure, we can notice that the performance of the model can be improved by using bigger frames when training the LSTM part of the DeepSpeech model. However, the performance of the model does not differ a lot – only a few percentage points.

The results show that some of the DeepSpeech architectures can perform better than the majority classifier but none of the architectures outperforms the baseline model. A possible explanation for this could be that these two tasks are simply not related enough and we cannot use information from the DeepSpeech model to improve the performance of a model for speech emotion recognition.

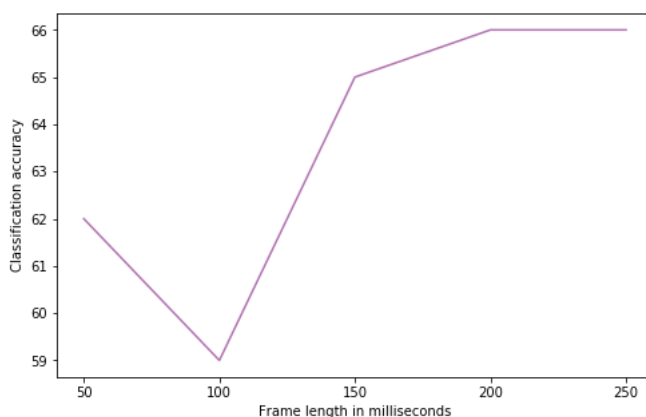


Figure 3: Performance of DeepSpeech model by using different frame lengths.

6 CONCLUSION

In this work we tried to improve a baseline speech emotion recognition classifier by transferring information from a pre-trained model. Although this transfer learning method has been most widely used in other computer science fields, most of the related work in speech emotion recognition developed transfer learning methods that transfer information from other emotional speech datasets into a target emotional speech dataset.

The pre-trained model we used was Mozilla’s DeepSpeech that was developed as a speech-to-text model. To recognize emotions in speech, we used the first layer from the DeepSpeech model, on top of which we added a new classifier that was trained from scratch on an emotional speech dataset. This way we repurposed the feature maps learned previously for the dataset.

The results from this approach did not seem to improve the classification accuracy of the improvisations part in the IEMO-CAP dataset. A possible explanation for this could be that the speech-to-text and speech emotion recognition tasks are simply not sufficiently related, and because of this the model could not extract any useful information from the DeepSpeech model. However, since this was the first attempt to transfer information from a well-defined pre-trained model to a speech emotion recognition task, we believe it is still a valuable attempt.

7 ACKNOWLEDGMENTS

This research has received funding from the European Union’s Horizon 2020 research and innovation programme under Grant Agreement No 769765

REFERENCES

- [1] Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N Chang, Sungbok Lee, and Shrikanth S Narayanan. 2008. Iemocap: interactive emotional dyadic motion capture database. *Language resources and evaluation*, 42, 4, 335.
- [2] Florian Eyben, Martin Wöllmer, and Björn Schuller. 2010. Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM international conference on Multimedia*, 1459–1462.
- [3] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. 2014. Deep speech: scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*.
- [4] Beth Logan et al. 2000. Mel frequency cepstral coefficients for music modeling. In *Ismir*. Volume 270, 1–11.
- [5] Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22, 10, 1345–1359.
- [6] Srinivas Parthasarathy and Carlos Busso. 2019. Semi-supervised speech emotion recognition with ladder networks. *arXiv preprint arXiv:1905.02921*.
- [7] Peng Song, Yun Jin, Li Zhao, and Minghai Xin. 2014. Speech emotion recognition using transfer learning. *IEICE TRANSACTIONS on Information and Systems*, 97, 9, 2530–2532.
- [8] Carl E Williams and Kenneth N Stevens. 1972. Emotions and speech: some acoustical correlates. *The Journal of the Acoustical Society of America*, 52, 4B, 1238–1250.

Towards Automatic Recognition of Similar Chess Motifs

Miha Bizjak

University of Ljubljana

Faculty of Computer and Information Science

Ljubljana, Slovenia

Matej Guid

University of Ljubljana

Faculty of Computer and Information Science

Ljubljana, Slovenia

ABSTRACT

We present a novel method to find chess positions similar to a given query position from a collection of archived chess games. Our approach considers not only the static similarity due to the arrangement of the chess pieces, but also the dynamic similarity based on the recognition of chess motifs and dynamic, tactical aspects of position similarity. We use information retrieval techniques to enable efficient approximate searches, and implement textual encoding that captures the position, accessibility and connectivity between chess pieces, pawn structures, and moves that represent the solution to the problem. We have shown experimentally how important the inclusion of both static and dynamic features is for the successful detection of similar chess motifs. In another experiment the program was able to quickly traverse a large database of positions to identify similar chess tactical problems. A chess expert found the resulting program useful for automatically generating instructive examples for chess training.

KEYWORDS

problem solving, chess motifs, automatic similarity recognition

1 INTRODUCTION

A significant part of acquiring human skills is to identify our weaknesses and take measures to remedy them. In problem-solving domains such as chess, the analysis of past games is important for players trying to improve their game. Identifying their mistakes enables chess players to work on improving some aspects of their game. This is often done by training on similar problems. Finding relevant similar problems involves recognising both static patterns, i.e. finding similar chess positions, and dynamic patterns, i.e. finding similar move sequences that solve a problem. These static and dynamic patterns are often referred to as *chess motifs*. Learning and recognising chess motifs during the game is one of the main prerequisites for becoming a competent chess player [2].

Chess instructors often look for examples containing relevant chess motifs from real games to provide their students with useful teaching material. However, it is impossible for a human being to go through thousands or even millions of games and find problem positions with similar chess motifs and similar solutions to those overlooked by the students in their game. Finding contextually similar chess positions could also be used for annotating chess games [5] and in intelligent chess tutoring systems [10].

The goal of our research is to develop a method to automatically retrieve chess positions with similar chess motifs for a given query position from a collection of archived chess games.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

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

© 2020 Copyright held by the owner/author(s).

1.1 Related Work

Existing chess search systems equipped with a query-by-example (QBE) [11] search interface are limited to searching only the exact matches in response to a given query position. To alleviate the problem of exact position searches, the Chess Query Language system (CQL) [1] allows the search for approximate matches of positions. However, it requires the user to define complex queries in the system-specific language. The search results can be sorted by any user-defined feature. In addition, the CQL works directly on game files and checks each game sequentially, making it inefficient for querying larger databases.

To overcome these problems, an approach has been proposed which is based on information retrieval for obtaining similar chess positions [4], constructing a textual representation for each board position and using information retrieval methods to calculate the similarity between these documents. Instead of constructing a query manually, the user specifies a chess position and a query encoding the characteristics of the position is automatically generated internally. Initially, a naive encoding was used, which only contains the positions of the individual pieces. The results have been improved by including additional information about the mobility of the individual pieces and the structural relationships between the pieces. Further work has been carried out to improve the quality of retrieval by implementing automatic recognition of pawn structures [7]. The additional information provided by the application of domain knowledge has proved useful, however, the positions are still only statically evaluated.

All existing approaches have a common shortcoming: they only allow the search for statically similar positions, while ignoring the dynamic factors, which are often far more important to obtain relevant search results.

2 DOMAIN DESCRIPTION

In this paper, we will focus on automatic retrieval of similar chess tactical problems from a large database of chess games. In chess, the term *tactic* is used to describe a sequence of moves that takes advantage of a certain position on the board and allows the player to gain material, a positional advantage, or even leads to a forced checkmate sequence.

Chess tactical problems are particularly important for the progress of chess players. Knowledge of tactical motifs helps them to quickly recognise the possible presence of a winning or drawing combination in a position. Chess players improve their tactical skills by solving tactical problems. A large number of games are decided by tactics, since a single mistake, which gives the opponent an opportunity for tactics can change the outcome of a game. To help players to discover tactical possibilities in games, many common patterns or *tactical motifs* have been defined in the chess literature [6]. Stoiljkovikj et al. developed a method for estimating the difficulty of chess tactical problems [9]. They introduced a concept of *meaningful search trees*, which can

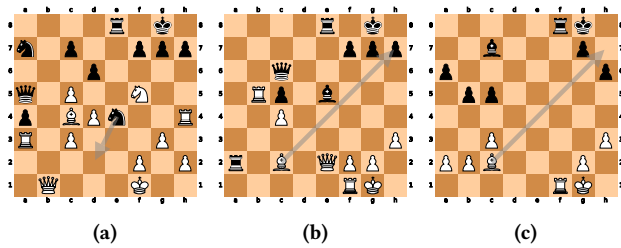


Figure 1: Tactical motifs.

potentially be used either for motif recognition or as an additional feature for positional similarity ranking.

We use standard chess annotation. Chess games are stored using Portable Game Notation (PGN), chess positions are described with Forsyth-Edwards Notation (FEN), and chess moves are described with Standard algebraic notation (SAN) [3].

Figure 1 shows some of the more common motifs. In Figure 1a, Black performs a *double attack* on the white king and queen at the same time. White must move the king out of check, allowing Black to capture the queen. Figure 1b is an example of a *discovered attack*. By moving the bishop, White opens the queen’s line of attack on the rook on a2. After Black responds to move out of the check, White can capture the black rook. The tactic in Figure 1c is called *deflection*. The black king protects the rook on f8. White gives a check with the bishop, forcing the black king to move away from the rook so that it can be captured.

To illustrate the difference between static and dynamic similarity using an example, we compare the query position in Figure 2a with the positions in Figure 2b and Figure 2c. The position in Figure 2b seems to be very similar to that in Figure 2a: only the white rook on h4 and the black rook on e8 have been removed. These two positions are statically similar. On the other hand, the position in Figure 2c seems to be quite different. However, if we compare the move sequences that represent solutions to these two tactical problems, we notice a great dynamic similarity. The solution in Figure 2a is 1. Rh8+ Kxh8 2. Qh6+ Kg8 3. Qxg7#. The solution in Figure 2c contains the same tactical motif as the solution mentioned above: the white rook is sacrificed on h8 and the black king must capture it, allowing the white queen to appear with check on h6 (note that it cannot be captured due to the activity of the white bishop along the long diagonal) and deliver checkmate on the next move. Note that such motif is not possible in the position shown in Figure 2b.

We are particularly interested in recognising the dynamic similarity, i.e. finding positions with similar motif(s) in the solution of the problem. However, we also want to take into account the static similarity, i.e. finding problems with similar initial position.

3 SIMILARITY COMPUTATION

To determine similarity between tactical problems we use an approach based on information retrieval. A set of features is computed from each problem’s starting position and its solution move sequence. The features are then converted into textual terms, forming a document that represents the problem. A collection of documents is used to build an index, which can then be queried using the textual encoding of a new position to retrieve the most similar positions in the index. For the implementation of the system for indexing and retrieval of similar tactics we use the *Apache Lucene Core* library. Search results are ranked using the BM25 ranking function [8].

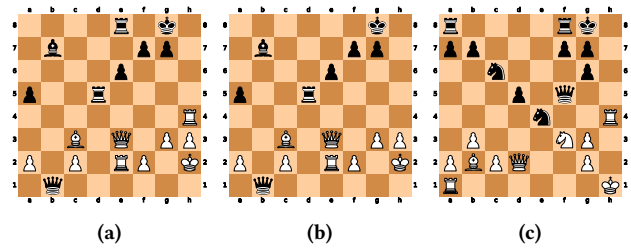


Figure 2: Static and dynamic similarity.

For each tactic, the input consists of a starting position in FEN format and a solution move sequence in algebraic notation. The solution can be provided with the position or calculated using a chess engine. Sections 3.1 and 3.2 describe the features and terms that are generated, and Figure 3 shows an example of a text encoding.

3.1 Static Features

The static part of the encoding includes information about the positions of pieces on the board, structural relationships between pieces and pawn structures present in the position.

The implementation is based on previous work on similar position retrieval [4] and pawn structure detection [7] and is intended to serve as a baseline on which we aim to improve by implementing encoding of dynamic features.

3.1.1 Piece positions and connectivity. The section describing piece positions and connectivity encoding consists of three parts:

- *naive encoding* - the positions of all the pieces on the board.
- *reachable squares* - all squares reachable by pieces on the board in one move, with decreasing weight based on distance from the original position, in format $\{piece\ symbol\ and\ position\}\{weight\}$.
- *connectivity between the pieces* - the structural relationships between the pieces in the positions. For each piece it is recorded which other pieces it attacks, defends or attacks through another piece (*X-ray attack*). Attacks are encoded as $\{attacking\ piece\ symbol\}\{attacked\ piece\ symbol\ and\ position\}$. For defense and X-ray attack terms, < and = separators are used instead.

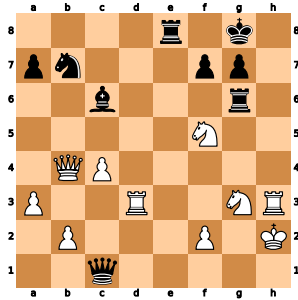
3.1.2 Pawn structures. For this section of the encoding, we use pawn structure detection algorithms [7] to detect the following pawn structures in the position and encode them into terms: isolated pawns ($I\{pawn\ position\}$), (protected) passed pawns ($F\{pawn\ position\}$), backward pawns, doubled pawns and pawn chains. Terms $P(\{number\})$ and $p(\{number\})$ are used to encode the number of pawn islands for white and black, respectively.

3.2 Dynamic Features

In the dynamic part of the encoding, we focus more on the solution of the tactical problem, trying to capture the motif behind it. We first encode some general characteristics of the solution, then add more specific terms describing the move sequence.

3.2.1 General dynamic features. In this part we encode some basic features of the solution move sequence that can help us determine similarity. We use a single term for each of the following features if it holds for the solution:

- $?px$ - the player captures a piece in at least one of the moves



(a) Encoded position. Black to play, solution: 1... Qh1+ 2. Nxh1 Rg2#.

Feature set	Generated terms
static_positions	qc1 Pb2 Pf2 Kh2 Pa3 Rd3 Ng3 Rh3 Qb4 ... qa1 0.78 qb1 0.89 qd1 0.89 qe1 0.78 ... q>Pb2 q>Pc4 Q>nb7 N>pg7 r>Ng3 P<Pa3 P<Ng3 K<Ng3 K<Rh3 P<Qb4 ... q=Pa3
static_pawns	If2 ia7 Fc4 P(2) p(2)
dynamic_general	?ox ?+ ?# ?S
dynamic_solution	!-q !-N !-r !-qN !-Nr !xq !Sq !#b !#r !#br !K>q !N>q !q>K !b>N !K>r !r>K !r>P

(b) Text encoding of each set of features for the above position.

Figure 3: Text encoding of a tactical position.

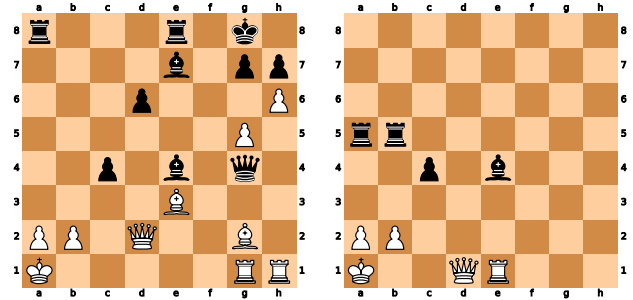
- ?ox - the opponent captures a piece in at least one of the moves
- ?+ - the player gives a check at least once during the sequence
- ?= - the player promotes a pawn in at least one of the moves
- ?S - the player sacrifices one or more pieces
- ?# - the solution ends with a checkmate
- ?1/2 - the solution ends in a draw

3.2.2 *Solution sequence features.* In this section we encode information about the solution move sequence. The encoding includes a term for each:

- type of piece moved: $!-\{piece\ symbol\}$
- type of piece captured: $!\{piece\ symbol\}$
- attack between pieces that occurs during the solution: $!\{attacking\ piece\ symbol\}\{attacked\ piece\ symbol\}$
- type of piece sacrificed: $!\{piece\ symbol\}$
- (if the final position is a checkmate) type of piece involved in checkmate: $!\#\{piece\ symbol\}$

We count a piece as involved in checkmate if it is attacking either the king directly or any of the squares where the king could move from the current position (ignoring checks).

To include information about the order of moves and captures we also include a term for each two consecutive moves and captures in the solution. We also include a term for each pair of pieces involved in checkmate to capture more specific combinations of pieces.



(a) Base problem. Black to play, solution: 1... Rxa2+ 2. Kxa2 Ra8+ 3. Ba7 Rxa7+ 4. Qa5 Kxa2 Ra5+ 3. Qa4 Rxa4#.

(b) Simplified problem. Black to play, solution: 1... Rxa2+ 2. Ra8+ 3. Ba7 Rxa7+ 4. Qa5 Kxa2 Ra5+ 3. Qa4 Rxa4#.

Figure 4: A pair of tactical problems from the data set.

4 EXPERIMENTAL RESULTS

To evaluate the effectiveness of our methods, we used a number of problems that we have collected from the Chess Tactics Art (CT-ART 6.0) training course¹. Many puzzles in this course consist of pairs of positions: one is taken from a real game, another represents a simplified version where the same tactical motif usually appears on a smaller 5×5 board. This fact allowed us to obtain a set of position pairs that were considered similar by human experts. We manually checked the puzzles and verified the similarity between the solutions of the individual problem pairs. A total of 400 pairs were collected for the test data set.

An example of such a pair is shown in Figure 4. The solution to both problems is to sacrifice the rook on the a-file to expose the king, resulting in checkmate with the other rook and the bishop on e4. The solution in the simplified problem contains the same motif, but there are much fewer pieces, so the solution is generally easier for the students to find.

4.1 Evaluation of Similarity Detection

We tested the effectiveness of our methods using the set of 400 pairs of problems described in the previous section. We first built an index using the simplified version of the problem from each pair, then performed a query on the index with each of the regular problems. For each query we recorded the rank of the matching position in the results and calculated how often the matching position appeared as the top result or within the first N results.

We tested the search accuracy using the following feature subsets: each feature group on its own, all static features, all dynamic features and all features combined. All runs used the default BM25 parameters $k_1 = 1.2$ and $b = 0.75$ and all included feature sets were weighted equally. The results are presented in Table 1.

Using either only static or dynamic features did not yield the best results. The results were significantly improved when both static and dynamic features were combined. This shows that each set of features covers a different aspect of a tactic, both of which need to be considered when determining similarity.

4.2 Similar Position Retrieval

In the second experiment, we selected 10 contextually different chess tactical problems and then automatically retrieved 5 most similar positions for each of them from a large database of 278,840

¹<https://chesskingtraining.com/ct-art>

Feature set used	Accuracy		
	top-1	top-5	top-10
static_positions	0.234	0.378	0.428
static_pawns	0.033	0.083	0.126
dynamic_general	0.008	0.038	0.071
dynamic_solution	0.421	0.657	0.761
all static features	0.252	0.370	0.433
all dynamic features	0.418	0.652	0.761
all features, equal weights	0.481	0.736	0.814

Table 1: Success rates for different configurations.

tactical problems constructed from the lichess.org game database. Building the index took about 14 minutes (it only needs to be done once), and retrieval was fast: only about 4 seconds.

Figure 5 shows a query position and the first two of the five most similar retrieved positions. This example illustrates how similarity ranking works and how the static and dynamic features contribute to the similarity scores of the results. The query position is an example of a discovered attack motif. With 1... Bh2+, Black sacrifices the bishop to later capture the rook on e1 with the queen. The first result shows the same motif with an almost identical move sequence. The main difference is that the key pieces are on the d-file and not on the e-file. The second result is another case of a discovered attack. In this example it is not a bishop but a knight sacrificed with a check to the white king. It is the static similarity (the arrangement and position of the pieces in the initial position) that contributes most to the great overall similarity of this tactical problem, although a certain dynamic similarity was also detected.

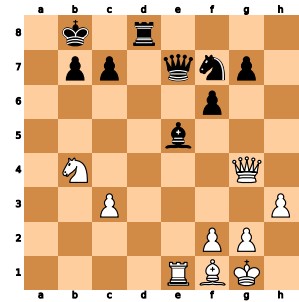
The resulting most similar positions were shown to a chess expert. The expert was asked to comment on the reasons for the similarity of the resulting problems with the original query positions, taking into account both static and dynamic aspects. The expert was able to explain the similarity in 48 out of 50 problems. Overall, the expert praised the program’s ability to detect dynamic similarity of positions, even if the initial positions differ significantly.

5 CONCLUSIONS

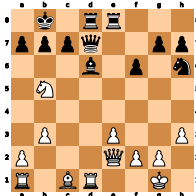
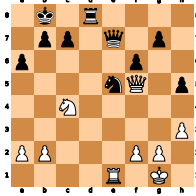
We introduced a novel method for retrieving similar chess positions, which takes into account not only static similarity due to the arrangement of the chess pieces, but also dynamic similarity based on the recognition of chess motifs and dynamic, tactical aspects of position similarity. The merits of the method were put to the test in two experiments. The first experiment emphasized the importance of including both static and dynamic features for the successful detection of similar chess motifs. In the second experiment, the program was able to quickly traverse a large database of positions to identify similar chess tactical problems. A chess expert was able to explain the similarity in the vast majority of the retrieved problems and praised the program’s ability to detect dynamic similarity of positions even if the initial positions differ significantly. The resulting program can be useful for the automatic generation of instructive examples for chess training.

REFERENCES

- [1] G Costeff. 2004. The Chess Query Language: CQL. *ICGA Journal*, 27, 4, 217–225.



(a) Query position. Black to play, solution: 1... Bh2+ 2. Kxh2 Qxe1.

Position	Solution	Similarity score
	1... Bh2+ 2. Kxh2 Qxd1	static 38.95 dynamic 45.04 total 83.99
	1... Nf3+ 2. Qxf3 Qxe1+	static 64.62 dynamic 12.32 total 76.94

(b) Retrieval results.

Figure 5: Example of retrieval results.

- [2] Mark Dvoretzky and Artur Yusupov. 2006. *Secrets of Chess Training*. Edition Olms.
- [3] International Chess Federation (FIDE). 2020. The FIDE Handbook. <https://handbook.fide.com/>. (2020).
- [4] Debasis Ganguly, Johannes Leveling, and Gareth JF Jones. 2014. Retrieval of similar chess positions. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. ACM, 687–696.
- [5] Matej Guid, Martin Možina, Jana Krivec, Aleksander Sadikov, and Ivan Bratko. 2008. Learning positional features for annotating chess games: A case study. In *International Conference on Computers and Games*. Springer, 192–204.
- [6] Chess Informant. 2014. *Encyclopedia of Chess Combinations, 5th Edition*. Chess Informant.
- [7] Matic Plut. 2018. Recognition of positional motifs in chess positions. Diploma thesis. University of Ljubljana.
- [8] Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. 1995. Okapi at trec-3. *Nist Special Publication Sp*, 109, 109.
- [9] Simon Stoiljkovikj, Ivan Bratko, and Matej Guid. 2015. A computational model for estimating the difficulty of chess problems. In *The Annual Third Conference on Advances in Cognitive Systems*.
- [10] Beverly Park Woolf. 2010. *Building intelligent interactive tutors: Student-centered strategies for revolutionizing e-learning*. Morgan Kaufmann.
- [11] Moshé M Zloof. 1975. Query-by-example: the invocation and definition of tables and forms. In *Proceedings of the 1st International Conference on Very Large Data Bases*, 1–24.

Drinking Detection From Videos in a Home Environment

Carlo M. De Masi
carlo.maria.demasi@ijs.si
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia

Mitja Luštrek
mitja.lustrek@ijs.si
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia

ABSTRACT

We present a pipeline developed with the aim of helping people with mild cognitive impairment (MCI) in the accomplishment of every-day tasks. Our system adopts a number of computer vision methods to analyze RGB videos collected from cameras, and provides a successful, quasi real-time detection of the targeted activity (drinking) when the latter is at least partially visible to the camera.

KEYWORDS

computer vision, activity recognition, object detection, pose estimation

1 INTRODUCTION

Mild cognitive impairment (MCI) is a common problem among elders, affecting 15–20% of people over 65 in the USA [10]. In order to help people affected by MCI in the accomplishment of every-day tasks, we adopt various kind of detection techniques to predict what users are currently doing, which, combined with a knowledge of their activities schedule, allows our system to provide context-based reminders. Here, we present our attempts to detect one of such activities (i.e. drinking) from videos, by the use of computer vision and deep learning algorithms.

This paper is organized as follows. In the remainder of this section, we give an overview of the current SOTA regarding activity recognition from videos. In Section 2 we describe the computer vision techniques used to trigger the more computationally intensive task of activity recognition, to obtain a quasi real-time monitoring of the user’s activities. Finally, in Sections 3 and 4 we present the results and conclusions of the paper.

1.1 Video Activity Recognition

Differently than what happened for image classification, where in the last years a number of clear front runner architectures and techniques have been established, the topic of activity recognition from videos still presents numerous open issues [1].

An immediate approach to the problem consists in using image classification networks to extract features from each frame of the video; then, predictions for the whole video can either be obtained by pooling over frames (at the cost of losing information about temporal ordering) [5], or by adopting LSTM layers [2].

A more elaborate way to adapt the concepts used in image classification methods to video recognition consists in using 3DCNN, i.e. convolutional models characterized by an additional third temporal dimension [4, 12, 13]. The increased number of

parameters makes 3DCNNs generally harder to train than their 2D counterparts. One way to fix this is to produce 3D models by "inflating" 2D ones, i.e. by adding a temporal dimension to a model pre-trained for image classification. This allows to determine the architecture of the 3D network and to bootstrap its values starting from the corresponding values in the 2D model: convolutional kernels with dimensions $N \times N$ are inflated to a 3D kernel with dimensions $N \times N \times t$, spanning t frames, and each of the t planes in the $N \times N \times t$ kernel is initialized by the pre-trained $N \times N$ weights rescaled by $1/t$ [1, 9].

Another approach separately analyzes spatial components (i.e. single frames), providing static information about scenes and objects in the picture, and temporal components related to motion and variation between frames [11]. A two-stream network parallelly processes single frames and optical flows, respectively, and then combines their predictions.

Finally, another method worth mentioning is based on the observation that some actions (i.e., clapping hands) are better characterized by high-frequency temporal features, whereas other ones (i.e., dancing) can be better understood when lower frequency variations are observed. As a result, a model characterized by two parallel channels can be used. The first (slow) channel operates at low framerate and analyzes few sparse frames, in order to deduce the semantics of the action, while the second (fast) branch is responsible for capturing fast variations, and so operates at higher framerate [3].

In this work, we adopted a modified version of an inflated 3D network as described in [14], to include non-local blocks. Unlike convolutional and recurrent operations, which are only able to capture spatio-temporal features in a local neighborhood, non-local blocks compute the response at a certain position as a weighted sum of features at all positions in space and time. This allows the model to capture dependencies between pixels that are distant both in space and time, and makes it more accurate for video classification.

2 SYSTEM ARCHITECTURE

The purpose of our system is to provide users context-based reminders related to the activity of drinking. To this aim, a RGB camera is placed in the kitchen of the user’s apartment (where the activity is most likely to take place) and the video is sent through a RTSP stream to a remote server, to be analyzed by the activity recognition model during the day. The results are uploaded to a Cloud Firestore Database, which is queried to determine whether the users have been drinking enough, and reminders are provided through an app running on a local device if not.

One problem arising from this scheme is that most action recognition algorithms are computationally expensive, which prevents them from running in real time. For this reason, we decided not to run the model continuously, but to execute it only in moments where it is most likely that the users are about to perform the targeted activity. We employed a combination of

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

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

© 2020 Copyright held by the owner/author(s).

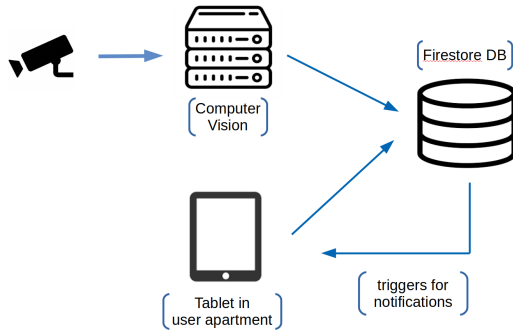


Figure 1: System architecture. Video stream from RGB cameras is sent to a remote server and fed to the activity recognition model. Results are uploaded to a Firestore database, where they are monitored so that notifications can be sent back to an app.

classic and deep-learning-based computer vision techniques to identify some *triggers* for the video activity recognition model, such as: (i) user standing in certain areas of the kitchen; (ii) user standing in certain areas of the kitchen, and interacting with some objects (tap, fridge); (iii) a specific object, assumed to be used by the user for drinking, is moved from its current position.

2.1 User Localization And Interaction With the Environment

The localization of the user and their interactions with the environment are detected through a combination of object detection and pose estimation techniques. For the object detection, we adopted a Single Shot MultiBox Detector (SSD) [8], pre-trained on the 80 classes of the COCO dataset [7], which also include "person". As for pose estimation, we used a SimpleNet model with a ResNet backbone [15].

During the initial setup, the camera image is shown to the user (Fig. 2a) and regions of interest (ROIs) can be selected (Fig. 2b). These can be of two types, i.e. single or double-zone. The first ones are identified by a single rectangular box, which is activated when the user's feet are within the box, hence providing indications on the user's location (see Fig 2c). Double-zone ROIs are formed by two rectangular boxes; one of them, analogously to the previous case, is activated when the user steps inside of it, while the second box is activated if one of the user's hands (located by the pose estimation model) is within it (Fig. 2d). Overall, a double-zone ROI is considered activated only if both conditions are met. Once the ROI is configured, the user is requested to input:

- the name used to identify the current ROI;
- an observation time t_{obs} (in seconds), i.e. the time after which the ROI is activated, once the requirements (user and hands positions) are met;
- an action to be performed once the ROI is activated. Currently, only one default action - recording and analyzing video clips - is supported, but this will be extended to include further possibilities.

2.2 Drinking Vessel Position Detection

A second trigger for activity recognition is given by the displacement of a particular object (mug, cup, glass). To this regard, in the pilot phase of the project users will be asked to always use one specific drinking vessel when they are drinking, which the model will be trained to recognize.

For this task, we considered two possibilities:

- a classic computer vision approach, where the drinking vessel is located through a color/shape-based detection;
- a deep learning object detection algorithm, re-trained to detect a personalized mug.

In the first scenario, we applied a series of filters (Gaussian blur, dilation/erosion) to reduce noise, followed by a color mask in the HSV space to select only objects with a certain color. A further selection is then done based on the shape properties of the previously selected areas; a polygonal approximation of their contours is performed, and other shape-related features such as area, circularity and convexity are considered to eliminate shapes different from the expected one.

In the second case, we collected a dataset of about 500 images of the selected mug, and used it to re-train a second SSD model. In order to account for false negatives in the mug detection, that may occur in some frames even if the mug has not been moved, for each frame the current position of the mug is compared to the history of positions in the past few frames. Once a displacement of the mug is detected, the trigger is activated.

2.3 Clip Recording and Activity Recognition

Following the activation of one of the triggers, the next video frames (for a time interval of about 30 seconds) are used to generate short video clips, each of which has a duration of 10 seconds, with an overlapping window of 4 seconds. These values have been selected to have a higher probability to obtain at least one video clip completely capturing the whole drinking process, and to match the length of the videos in the Kinetics400 dataset [6], which has been used for the activity-recognition model training.

3 RESULTS AND DISCUSSION

In this section, we present the results of the various steps involved in the whole drinking-detection pipeline.

3.1 User Localization - Results

We tested the efficiency of the localization module in different scenarios, varying based on how clearly the user was visible (completely visible; legs occluded; head occluded; head and legs occluded, only torso visible) and on which side (front/back/right/left) of the user was visible, and the results showed an average accuracy of over 98%.

3.2 Drinking Vessel Position Detection - Results

As illustrated in Sec. 2.2, for the task of detecting the displacement of the drinking vessel we adopted two approaches, one based on classic computer vision methods and one on deep learning.

The first method does not provide a confidence score for detections, nor the coordinates of the object's bounding box, so we took a simpler approach than with normal object detection algorithms in evaluating the results. We collected some videos in a home-like environment, with the object located in different positions, or with a person handling it (moving it, using it to drink...), and analyzed them frame-by-frame to check whether the objects present in each frame were detected or not. The resulting confusion matrix, reported in Table 1, shows that the detection algorithm scored precision and recall values of .93 and .90, respectively. This method proved to be very efficient, when correctly fine-tuned, and the algorithm detected the object in most of the frames where it was at least partially visible. The

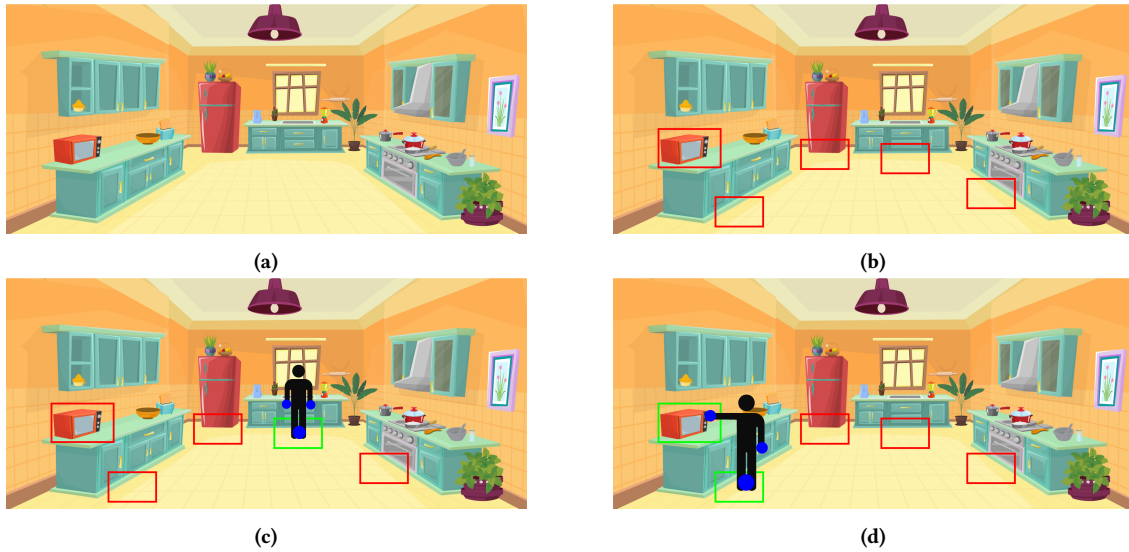


Figure 2: Triggers based on user's location and their interaction with the environment. Regions of Interest are selected during the setup phase (b), and they are activated either if the user steps inside (c), or if the user steps inside and has their hands next to another object (d).

Table 1: Confusion matrix for the color/shape-based detection of the mug

		Pred	
		P	N
True	P	133	15
	N	10	1

greatest issue of the method is that it had to be very carefully tuned, especially regarding the color selection part, which is still sensible to lightning variations even after converting the image to the HSV colorspace. False detection can also be a problem. We tested the algorithm in situations where some of the objects present in the scene had colors similar to the object we wanted to detect, and in spite of being able to filter out most of them we still obtained some false positives, especially when the lighting varied, thus rendering the selection of the parameters for the color mask less efficient.

The results of the evaluation of the SSD model are shown in Fig. 3. As evident from the plot, the model immediately reached a very high mAP [7], of the order ≈ 0.9 , on our test dataset. It should be noted that, while preparing the training dataset, we followed a somewhat different approach than what is usually done for training object-detection models. In most situations, one wants to make the model as general as possible and avoid overfitting, which is achieved by taking images of the desired object in as many different conditions (size, aspect ratio, point of view angle, rotation, lightning) as possible. In our case, however, the location of the camera will be more or less constant, i.e. attached to the ceiling of the room, in order to provide a good view of the environment. As a result, this will greatly limit the variability in the images of the object the system will analyze, especially regarding the aspect ratio and the orientation of the mug. Moreover, whereas an object detector is usually tasked to identify many different instances of objects in a certain class (i.e., a generic "mug"), in our case the task is greatly simplified by the fact that we are looking to locate one very specific object.

3.3 Activity Recognition - Results

We tested the adopted activity recognition model on a new custom dataset, consisting of roughly 100 videos we recorded ourselves in a variety of environments and conditions. In order to make the clips as similar as possible to real-life situations, the videos contained instances where actions similar to drinking were performed, to increase the recognition difficulty. The clips can be classified as belonging to two difficulty categories, based on the angle the user was facing with respect to the camera; videos were classified as "hard" whenever this angle was greater than 90° (see Fig. 4). The precision-recall curve for the mug on this dataset is shown in Fig. 5.

4 CONCLUSIONS

The tests performed on triggers are very encouraging for the one based on the user location and their interaction, and indicate that the deep-learning approach should be preferable for the detection of the drinking vessel and its displacement, especially after increasing the amount of training data. The activity-recognition model based on inflated 3D CNN with the addition of non-local blocks provided the best accuracy in situations where the user is facing the camera at least partially, and the use of triggers allows for a quasi real time usage. A number of improvements will be added to the pipeline in the future. Currently, only one action is triggered, i.e. recording and analysis of video clips, but we plan to include other possibilities, such as using the information on the user location to check whether they need assistance in operating domestic appliances. The object detection model could also be extended, in order to identify interactions with other elements of the environment, and provide corresponding context-based responses. Finally, the only action currently recognized is drinking, but as mentioned in the introduction the aim of the project is to assist users in the accomplishment of various activities. In this sense, the next planned step is to include detection of parts of the morning toilet routines, such as brushing teeth and washing hands.

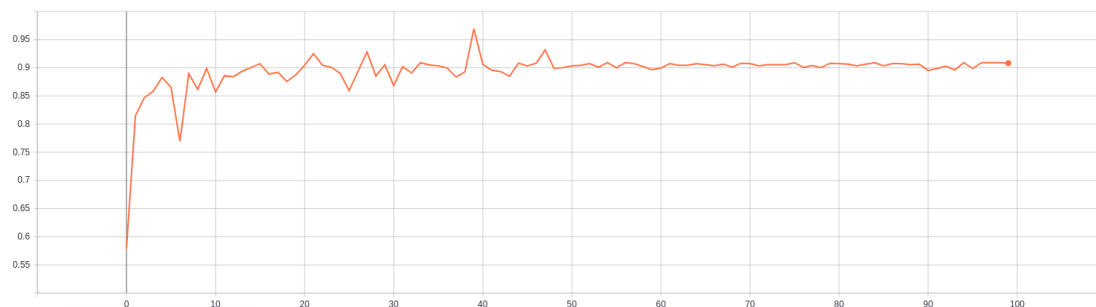


Figure 3: mAP values on the test dataset for the SSD model, re-trained to recognize the project custom mug.

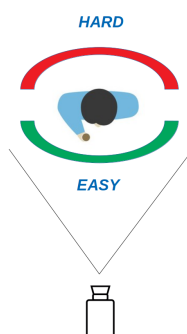


Figure 4: Difficulty classes for the custom dataset we used to test the activity recognition model. Video clips were classified as "hard" whenever the angle between the user front side and the camera was greater than 90° .

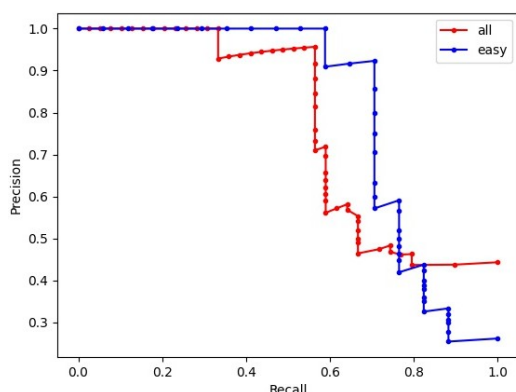


Figure 5: Test results of the activity recognition model on the test dataset.

REFERENCES

- [1] Joao Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 6299–6308.
- [2] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, et al. 2015. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2625–2634.
- [3] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, et al. 2019. Slowfast networks for video recognition. In *Proceedings of the IEEE international conference on computer vision*, 6202–6211.
- [4] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 2012. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35, 1, 221–231.
- [5] Andrej Karpathy, George Toderici, Sanketh Shetty, et al. 2014. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 1725–1732.
- [6] Will Kay, Joao Carreira, Karen Simonyan, et al. 2017. The kinetics human action video dataset. (2017). arXiv: 1705.06950 [cs.CV].
- [7] Tsung-Yi Lin, Michael Maire, Serge Belongie, et al. 2014. Microsoft coco: common objects in context. (2014). arXiv: 1405.0312 [cs.CV].
- [8] Wei Liu, Dragomir Anguelov, Dumitru Erhan, et al. 2016. Ssd: single shot multibox detector. *Lecture Notes in Computer Science*, 21–37. ISSN: 1611-3349. DOI: 10.1007/978-3-319-46448-0_2. http://dx.doi.org/10.1007/978-3-319-46448-0_2.
- [9] Elman Mansimov, Nitish Srivastava, and Ruslan Salakhutdinov. 2015. Initialization strategies of spatio-temporal convolutional neural networks. *arXiv preprint arXiv:1503.07274*.
- [10] Ronald C Petersen, Oscar Lopez, Melissa J Armstrong, et al. 2018. Practice guideline update summary: mild cognitive impairment: report of the guideline development, dissemination, and implementation subcommittee of the american academy of neurology. *Neurology*, 90, 3, 126–135.
- [11] Karen Simonyan and Andrew Zisserman. 2014. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, 568–576.
- [12] Du Tran, Lubomir Bourdev, Rob Fergus, et al. 2015. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, 4489–4497.
- [13] Gül Varol, Ivan Laptev, and Cordelia Schmid. 2017. Long-term temporal convolutions for action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40, 6, 1510–1517.
- [14] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. 2018. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7794–7803.
- [15] Bin Xiao, Haiping Wu, and Yichen Wei. 2018. Simple baselines for human pose estimation and tracking. In *Proceedings of the European conference on computer vision (ECCV)*, 466–481.

Semantic Feature Selection for AI-Based Estimation of Operation Durations in Individualized Tool Manufacturing

Erik Dovgan
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia
erik.dovgan@ijs.si

Bogdan Filipič
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia
bogdan.filipic@ijs.si

ABSTRACT

Accurate estimation of operation durations is of key importance in production processes, since the accuracy of estimations directly affects the quality of production plans and thus the entire production process. This task is even more challenging when individualized tools are being produced. From the machine learning point of view, this means a low number of diverse samples, while the number of features can be significantly higher. To tackle this issue, we introduce semantic feature selection that reduces the number of features. This results in obtaining a better ratio between the number of samples and features and, at the same time, reduces the prediction error. We demonstrate the proposed approach on the problem of estimating the operation durations in the manufacturing of injection molds and show the prediction accuracy improvement resulting from the semantic feature selection.

KEYWORDS

injection molding, tool manufacturing, duration prediction, feature selection, random forest

1 INTRODUCTION

The efficiency of tool shop manufacturing processes heavily depends on the accuracy of production plans. Inaccurate plans can lead to significant delays in production, due date violations, late delivery penalties, and even loss of customers. A key step of planning is accurate estimation of durations of all the operations to be executed in the manufacturing process. The estimation can be performed manually by an expert utilizing his/her expert knowledge, or automatically by means of tools such as those involving AI methods as, for example, demonstrated in [3].

Automated estimation of operation durations with AI methods consists of learning a predictive model from the features extracted from examples of past, i.e., already concluded operations and their actual durations, and then applying the model to new operations with known features and unknown durations. In the case of tool manufacturing, the features can be extracted from 3D computer models of already manufactured tools. To build an accurate predictive model, a large set of already manufactured tools has to be processed. However, this is not possible in certain cases, for example, when dealing with individualized tools, such as injection molds. This is due to the fact that the tool shops specialized in individualized tool manufacturing typically produce only few such tools per year. In addition, these tools are

very diverse, which increases the difficulty of automated duration prediction.

We propose an approach for predicting operation durations in the manufacturing of individualized tools. The tools are manually divided into several positions of varying complexity, where each position is specified with a 3D computer model. In addition, a set of operations are predefined for each of these positions. The proposed approach processes the 3D model of each position and predicts the duration of the corresponding manufacturing operations. To this end, it firstly extracts a set of volume, surface, gradient and other features from the 3D model, and then applies the Random Forest regression model [1] to predict the duration of each operation. This process is additionally enhanced with semantic feature selection that evaluates various sets of semantically related features, such as volume features, in order to assess the predictive capability of these feature sets. We demonstrate the proposed approach on the problem of estimating the operation durations in the manufacturing of injection molds in a specific tool shop. By processing a dataset from this tool shop, we show the prediction accuracy improvement resulting from the semantic feature selection.

The rest of the paper is organized as follows. Section 2 introduces the relevant tool positions and the related operations, and describes the extracted features and the semantic feature selection. Numerical experiments and the obtained results are presented in Section 3. Finally, Section 4 concludes the paper with the summary of our work and the ideas for future work.

2 PREDICTING OPERATION DURATIONS WITH AI METHODS

Prediction of operation durations consists of extracting features from the tool data in the form of 3D computer models, and applying a machine learning model to predict the durations. This approach is applied for each tool position and each operation at this position independently, thus a custom machine learning model is built and applied for each combination of position and operation. In addition, when feature selection is involved, a different set of features is considered for each of these combinations.

2.1 Relevant Positions and Related Operations

The tools regarded in this study are injection molds that are used to form the final products made of plastic under high pressure. Although the injection mold is composed of several positions, its most complex and thus the most relevant positions are the bottom and the top element. These two elements have to be manufactured with the highest precision. Since they are in physical contact with the final product, any defect of the mold surface would result in a defect of the final product. An example of the injection mold is shown in Figure 1, where the red color indicates the surface

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

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

© 2020 Copyright held by the owner/author(s).

that is in contact with the final product. In the dataset used in this study, these two elements are marked as positions 1 and 30. These positions require a set of operations, where the most relevant operations are shown in Tables 1–2.

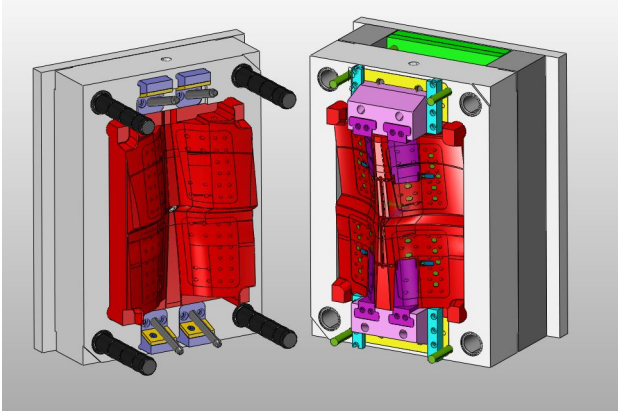


Figure 1: Example of a 3D computer model of an injection mold, <https://grabcad.com/library/injection-mold-pc-abs-1> by Mauro Menchini.

Table 1: Operations at Position 1

Operation	Description
32	CAM rough
31	CAM fine
43	CAM erosion
19	Heat treatment
23	Measuring machine
36	CNC milling 3 axis, rough
41	CNC milling 3 axis, fine
42	CNC milling 5 axis, fine
13	Submersible erosion

Table 2: Operations at Position 30

Operation	Description
32	CAM rough
31	CAM fine
37	CAM wire erosion
43	CAM erosion
19	Heat treatment
11	Wire erosion
23	Measuring machine
36	CNC milling 3 axis, rough
41	CNC milling 3 axis, fine
42	CNC milling 5 axis, fine
13	Submersible erosion

2.2 Description of the Extracted Features

The proposed approach extracts a set of features from a 3D computer model of a tool. These features were suggested by a tool shop expert and can be categorized as follows:

- volumes of the entire tool position (such as volume of the shape and volume of the mold),
- volumes of the holes that are open, and of those that are closed,
- features for each of 6 directions, i.e., projections (x, y, z, each of them increasingly or decreasingly), for example, direction (z, decreasingly) defines the features obtained from the top-down projection, while direction (z, increasingly) defines the features obtained from the bottom-up projection; the features for each direction are as follows:
 - volumes (including the volumes of holes),
 - surface area,
 - number of faces,
 - number of faces per dm^2 ,
 - valley features, computed as the height versus width ratio of the valleys (in all valley directions to find the maximum value); this feature is aimed at identifying deep and narrow valleys that are harder to process,
 - valley height, computed as the height of the valleys in all valley directions to find the maximum value; this feature is aimed at obtaining the depth of valleys that represents the drill distance,
 - gradient features, calculated as the maximum gradient in all directions; this feature is aimed at identifying areas with non-horizontal and non-vertical gradient that are harder to process.

Since the valley features, valley height and gradient features are calculated for each point of the projection, the number of features is very high and varies across the tool positions which are of varying sizes. To reduce the number of features and obtain a constant number of features independently of the position size, histograms of these features are calculated using expert-defined bins.

The 3D model of each position also contains expert-defined annotations of the model parts with different colors of model faces (see the example in Figure 1). These model parts are also taken into account when extracting features and therefore obtaining additional features that characterize a feature for each part independently. For example, when calculating the number of faces, one feature is obtained for all the faces, and for each part an additional feature is calculated denoting the number of faces on that specific part. The part-specific features are calculated for the following features:

- volumes of the holes: total, open, closed,
- projection features:
 - volume,
 - surface area,
 - number of faces,
 - number of faces per dm^2 ,
 - valley features,
 - valley height,
 - gradient features.

Examples of parts that are annotated in the 3D computer models include: (1) Free holes, (6) Tolerance holes, (7) Parting surface, (10) Matching surfaces, (12_4) Part shape: High gloss polished, (12_5) Part shape: Optical faces, (12_7) Part shape: Galvanic pins, (12_8) Part shape: Special surface finishing. In total, 30 parts are annotated by the expert.

Table 3: Feature Sets

Name	Number of features
expert	524 on average
volume	6
volume_projection	30
volume_no_hole	3
volume_projection_no_hole	6
volume_hole	3
volume_projection_hole	24
volume_hole_part	90
volume_projection_no_hole_part	180
material	4
surface_projection	6
surface_projection_part	180
faces_count_projection	6
faces_count_projection_part	180
faces_per_dm2_projection	6
faces_per_dm2_projection_part	180
valley_hist_projection	18
valley_hist_projection_part	540
valley_h_projection	48
valley_h_projection_part	1440
grad_hist_projection	18
grad_hist_projection_part	540
projection_*	562
projection_side	2248
projection_top_bottom	1124
part_*	111

2.3 Semantic Feature Selection

The total number of features obtained in the presented feature extraction procedure is 3472. Since this is a large number, we introduce semantic feature selection that combines semantically similar features into (partially overlapping) feature sets. In addition, the tool shop expert also selected a set of the most relevant features for each operation. However, this was defined only for a limited set of crucial operations. The resulting feature sets and the related numbers of features are shown in Table 3. Specifically, if the name of a set contains "part", the set contains all the features of the specific part. The "valley_hist_" contains the valley features, "valley_h_" valley height, and "grad_hist_" gradient features. Projection sets "projection_" contain all the features from specific projections and are defined as follows:

- projection_100: projection from left to right (x axis)
- projection_200: projection from right to left (x axis)
- projection_010: projection from front to back (y axis)
- projection_020: projection from back to front (y axis)
- projection_001: projection from bottom to top (z axis)
- projection_002: projection from top to bottom (z axis)

In total, 60 sets of features were defined.

3 EXPERIMENTS AND RESULTS

We evaluated the proposed approach on a dataset from the Plamtex tool shop [4, 2]. Due to individualized tool manufacturing, the number of already produced tools was low, namely 30 instances of position 1 and 26 instances of position 30. Besides the actual duration of each operation, each instance also included the duration estimated by the tool shop expert.

The operation durations were predicted with the Random Forest regression model. Its performance was assessed with the leave-one-out test using the default model-building parameters. The selected performance metric was the Root Mean Squared Error (RMSE), which has to be minimized. RMSE was also calculated for durations estimated by the expert. The effectiveness of feature selection was determined by comparing the Random Forest performance when using all the features and when using only a selected set of features.

The initial experiment aimed at finding whether the prediction of operation durations involving the proposed feature selection outperforms the prediction without feature selection considering all the features (i.e., the default feature set). To this end, for each combination of position and operation, all the feature sets were processed and the feature set with the lowest RMSE was selected. The results are shown in Figure 2. These results are normalized with respect to the RMSE of durations estimated by the expert and are therefore expressed as percentages of the RMSE resulting from the expert estimation. They show that for each combination of position and operation, there exists at least one set of features that allows for more accurate prediction than the default feature set (since it reduces the RMSE). In addition, for position 1, operation 32, and position 30, operation 31, the default feature set produces a RMSE equal to the RMSE of the expert estimation, while feature selection improves it. For position 30, operation 32, the default feature set results in a higher RMSE than the RMSE of the expert estimation. Although in this case feature selection improves the result, it still performs worse than the expert estimation.

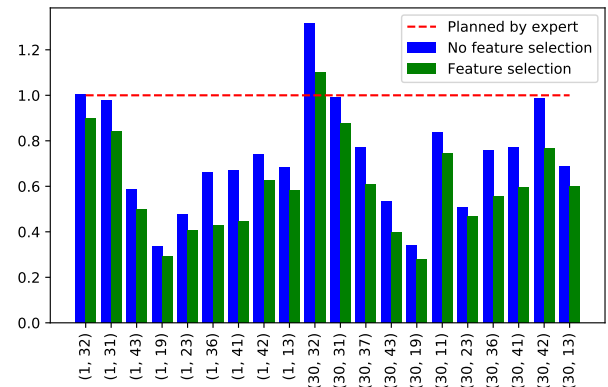


Figure 2: Percentages of RMSE with respect to the RMSE of durations estimated by the tool shop expert. The horizontal axis denotes the combinations of (position, operation).

Subsequently, the most relevant combinations of positions and operations were analyzed in more detail and selected results are presented in Figures 3–5. These results show the RMSE of durations estimated by the expert, the RMSE obtained without feature selection, and the RMSE obtained with various sets of features. To make the figures readable, we only show the best 33% feature sets. Figure 3 shows position 1 and operation 36 (i.e., CNC milling 3 axis, rough). The best features are the gradient features, surface features and features from the bottom-up projection. Note also that the bottom side of this position is the most complex one, thus the bottom-up projection is of high importance. The same projection is also the most relevant for position 1, operation 13 (i.e., submersible erosion) (see Figure 4), since

the erosion is applied only to the bottom side of this position. Part 9 (i.e., released surfaces) and faces count are also among the most important features, where faces count can be used to estimate the complexity of the surface that has to be eroded. Finally, position 30, operation 13 (i.e., submersible erosion) is shown in Figure 5. For this combination, the top-down projection is the most relevant, since the erosion is applied only to the top side of this position. Part 1 (i.e., free holes) and faces count are also very important. The importance of the appropriate projection and the faces count is consistent with the results for position 1 and the same operation (see Figure 4).

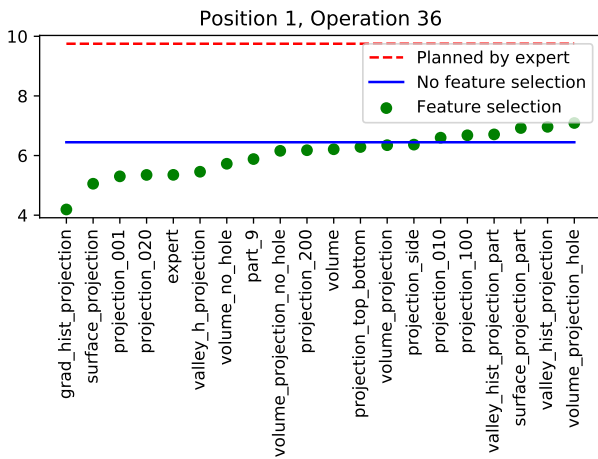


Figure 3: RMSE obtained when predicting the duration of operation 36 (CNC milling 3 axis, rough) at position 1.

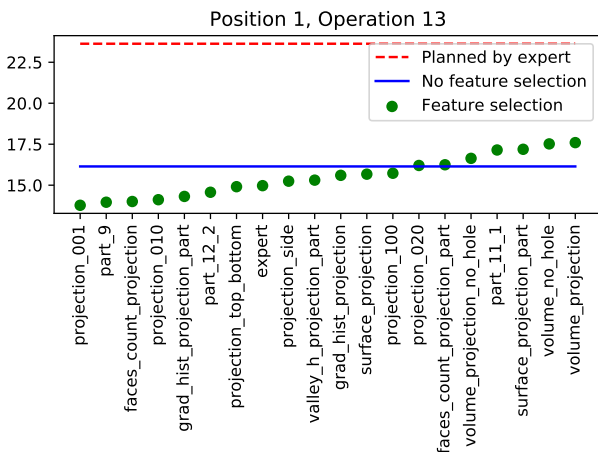


Figure 4: RMSE obtained when predicting the duration of operation 13 (submersible erosion) at position 1.

4 CONCLUSION

We presented an AI-based approach to predicting the operation durations in individualized tool manufacturing, which is, in a long run, aimed at replacing the existing human-based estimation process. The proposed approach extracts a set of features from 3D computer models of tools and applies Random Forest regression to predict the operation durations. To further improve

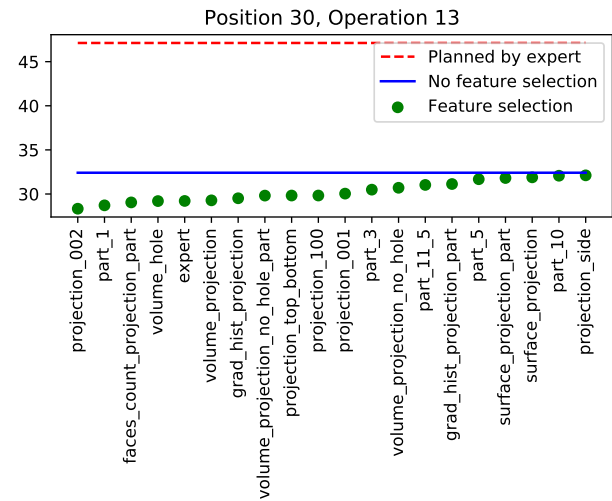


Figure 5: RMSE obtained when predicting the duration of operation 13 (submersible erosion) at position 30.

the prediction accuracy, it includes semantic feature selection by combining features into semantically meaningful feature sets. The experimental results showed that this approach in most cases outperforms the expert predictions. In addition, semantic feature selection outperforms the approach with no feature selection. A detailed analysis of the proposed feature selection approach showed that there exist meaningful relations between the tool manufacturing operations and the best performing feature sets for predicting the durations of these operations.

In future work we will evaluate additional regression algorithms to assess the quality of Random Forest predictions. It would be also relevant to analyze the samples for which the prediction error is the highest. Special attention should be given to the operation for which the presented approach did not outperform the expert prediction.

ACKNOWLEDGMENTS

This work was in part funded by the KET4CleanProduction project "Improved Planning of Manufacturing Processes for Individualized Tools" where the AI-based solution was developed for the Plamtex tool shop. The authors also acknowledge the financial support from the Slovenian Research Agency (research core funding No. P2-0209). We are particularly grateful to Plamtex for sharing the tool dataset and the expert knowledge on tool manufacturing, positions, operations, and the suitable features.

REFERENCES

- [1] Leo Breiman. 2001. Random forests. *Machine Learning*, 45, 1, 5–32.
- [2] Erik Dovgan, Peter Korošec, and Bogdan Filipič. 2020. Tool-Analysis: A program for predicting the duration of machining operations in the production of tools using artificial intelligence. Technical report IJS-DP 13195. Jožef Stefan Institute, Ljubljana.
- [3] Mesut Kumru and Pinar Yildiz Kumru. 2014. Using artificial neural networks to forecast operation times in metal industry. *International Journal of Computer Integrated Manufacturing*, 27, 1, 48–59.
- [4] Plamtex INT, d.o.o. 2020. <https://www.plamtex.si/en/>.

Generating Alternatives for DEX Models using Bayesian Optimization

Martin Gjoreski
Department of Intelligent
Systems

Jožef Stefan Institute
Jožef Stefan Postgraduate School
Ljubljana, Slovenia
martin.gjoreski@ijs.si

Vladimir Kuzmanovski
Department of Computer Science
Aalto University, Finland
vladimir.kuzmanovski@aalto.fi

Department of Knowledge
Technologies
Jožef Stefan Institute
Ljubljana, Slovenia

Marko Bohanec
Department of Knowledge
Technologies
Jožef Stefan Institute
Ljubljana, Slovenia
marko.bohanec@ijs.si

ABSTRACT

Multi-attribute decision analysis is an approach to decision support in which decision alternatives are assessed by multi-criteria models. In this paper, we address the problem of generating alternatives: given a multi-attribute model and an alternative, the goal is to generate alternatives that require the smallest change to the current alternative to obtain a desirable outcome. We present a novel method for alternative generation based on Bayesian optimization and adapted to qualitative DEX models. The method was extensively evaluated on 42 different DEX decision models with a variable complexity (e.g., variable depth and variable attribute's weight distribution). The method's behavior was analyzed with respect to computing time, time to obtaining the first appropriate alternative, number of generated alternatives, and number of attribute changes required to reach the generated alternatives. The experimental results confirmed the method's suitability for the task, generating at least one appropriate alternative within one minute. The relation between the decision-model's depth and the computing time was linear and not exponential, which implies that the method is scalable.

KEYWORDS

multi-attribute models, method DEX, alternatives, decision support, Bayesian optimization

1 INTRODUCTION

Hierarchical multi-attribute models are a type of decision models [1],[2],[3], which decompose the problem into smaller and less complex subproblems and represent it by a hierarchy of attributes and utility functions. Such decision models are especially useful in complex decision problems [4],[5].

DEX is a hierarchical qualitative multi-attribute method whose models are characterized by using qualitative (symbolic) attributes and decision rules. The method is supported by DEXi [6],[6],[7],[8], an interactive computer program for the

development of qualitative multi-attribute decision models and the evaluation of alternatives (options). DEXi has been used to analyze decision problems in different domains in healthcare [9], agriculture [10], [11], [12], economy [13], etc.

A useful extension of DEX would be the possibility to search for new alternatives that require the smallest change to the existing alternative to obtain a desirable outcome. This task is important for practical decision support [14], however the related work on generating alternatives for qualitative multi-attribute decision models is quite scarce. The only related study was presented by Bergez [15], in which the focus is on attribute scoring (and not on the alternatives), and the starting (current) alternative was not taken into a consideration. More specifically, Bergez developed a genetic algorithm for searching a set of the "worst-best" i.e., lowest scores for the input attributes that lead to the highest score for the root attribute (the decision model's output), and "best-worst" i.e., highest scores for the input attributes that lead to the lowest score for the root attribute.

In this study, we developed a stochastic method for generating alternatives that require the smallest change to the current alternative to obtain a desirable outcome. To avoid combinatorial explosion, the method uses guided search based on Bayesian optimization. The method is evaluated on 42 different qualitative multi-attribute models with a varying complexity. The method's behavior was analyzed with respect to several characteristics including: computing time, time to first appropriate alternative, number of generated (appropriate) alternatives, and number of attribute changes required to reach the generated alternatives.

2 DOMAIN DESCRIPTION

In this study, a set of 42 DEX multi-attribute decision models were used. The models are benchmark mock models, designed by Kuzmanovski et al. [16]. The decision models are designed by taking into account properties such as model depth, distribution of attributes' aggregation weights (weights' distribution), and inter-dependency of attributes (input links). Table 1 presents a summary of the decision models. The weights' distribution is given with descriptive names: skewed, normal, and uniform. All the attributes in the models are defined with same value scale (low, medium, high), including the input and the output attributes. Additional assumption is that all attribute combinations are possible.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
Information Society 2020, 5–9 October 2020, Ljubljana, Slovenia
© 2020 Copyright held by the owner/author(s).

Table 1: Properties of the mock DEX decision models.

Leaves	Depth	Weights' distribution	Links	Versions
8	3	skewed, normal, uniform	yes	3, 3, 1
9	3	skewed, normal, uniform	no	3, 3, 1
19	4	skewed, normal, uniform	yes	3, 3, 1
20	4	skewed, normal, uniform	no	3, 3, 1
38	5	skewed, normal, uniform	yes	3, 3, 1
39	5	skewed, normal, uniform	no	3, 3, 1

3 METHOD FOR GENERATING ALTERNATIVES

An efficient search strategy is required to generate alternatives that require the smallest change to the current alternative to obtain a desirable outcome. A naïve approach would be to generate all possible alternatives, or to iteratively generate random alternatives, and to evaluate the outcome for each alternative. However, for reasonably complex decision models, the search space can be enormous, rendering the naïve approaches unsuitable.

A more appropriate approach would be to use informed search based on the history of previously generated and evaluated alternatives. The history can be used to estimate the search space and the behavior of the decision model. Based on that estimation, more promising alternatives can be generated. By focusing on the more promising alternatives the search space is reduced, and consequently, the time needed to find the appropriate alternatives is also reduced. The next subsections describe a stochastic method that uses Bayesian optimization to efficiently generate such alternatives. The method assumes that we do not know the internal rules by which the decision models operate, thus it falls into the category of “black-box” optimization techniques. Knowing and utilizing the decision rules might help the search algorithm, but this option was not addressed in this study.

3.1 Implementation

The problem of generating alternatives that require the smallest change to the current alternative to obtain a desirable outcome can be defined as an optimization problem with two objectives: (1) improved outcome (desired output) of the decision model, and (2) maximum similarity between the current alternative \bar{c} , and the new proposed alternative \bar{a} . For each decision model DM , one alternative can be defined as a tuple of attributes $\bar{a} = (a_1, a_2, \dots, a_n)$, where each attribute can take any value of a limited set of values. Usually, that set includes ordinal values (e.g., low, medium and high) and those values can be encoded with integers (e.g., 0, 1 and 2). Consequently, a distance d between alternatives can be defined over Euclidean space. The specific distance function used by the method is a modified element-wise difference between the candidate alternative \bar{a} and the current alternative \bar{c} . This distance considers only the attributes for which the candidate alternative has higher values compared to the current alternative \bar{c} .

$$d(\bar{c}, \bar{a}) = \sum \begin{cases} a_j - c_j, & \text{if } a_j > c_j \\ 0, & \text{if } a_j \leq c_j \end{cases}$$

From the distance function, a similarity function s can be also defined as one minus the normalized distance. The distance is normalized using the maximum plausible distance for the specific problem. For example, if \bar{a} has 20 attributes with possible values between 0 and 2 and each attribute has the highest possible value, and if \bar{c} has only attributes with the lowest possible value (0), then the maximum distance is $20 * 2$.

$$s(\bar{c}, \bar{a}) = 1 - \frac{d(\bar{c}, \bar{a})}{\text{max_distance}}$$

Finally, the optimization function can be defined as:

$$f(\bar{c}, \bar{a}, DM(\bar{c}), DM(\bar{a})) = \begin{cases} s(\bar{c}, \bar{a}), & \text{if } DM(\bar{a}) > DM(\bar{c}) \\ 0, & \text{if } DM(\bar{a}) \leq DM(\bar{c}) \end{cases}$$

where $DM(*)$ is the output of the decision model for the specific alternative. By optimizing f , the method searches for alternatives that are as similar as possible to \bar{c} and improve the output of the decision model ($DM(\bar{a}) > DM(\bar{c})$).

In order to apply the Bayesian optimization approach, a surrogate function (a model), an acquisition function, and a generator of alternatives, need to be defined. The surrogate model SM is a model that estimates the objective function for a given alternative as input. Typically, models based on Gaussian Process (GP) [17] are used because by exploiting the mean and the standard deviation of the output distribution, we can balance the trade-off of exploiting (higher mean) and exploring (higher standard deviation). Since GP models are computationally expensive with the complexity of $O(n^3)$, ensemble models such as Random Forest (RF) can be also used [18]. In that case, the mean and the variance are calculated based on the predictions of all base models available in the ensemble. Our method uses RF with 1000 decision trees as base models.

The acquisition function operates on top of the mean and standard deviation of the SM 's output. The final version of the method uses the expected improvement (EI) as an acquisition function [19]. This acquisition function checks the improvement that each candidate alternative brings with respect to the maximum known value ($\mu(SM(\bar{a})) - a_b$), and scales those improvements with respect to the uncertainty. If two alternatives have a similar mean value, the one with higher uncertainty ($\sigma(SM(\bar{a}))$) will be preferred by the acquisition function.

Finally, we need to define the generator of alternatives. Our method uses two generators of alternatives: a neighborhood generator and a random generator. Based on the distance function d , neighborhood relation can be defined. Two alternatives \bar{a}_1 and \bar{a}_2 are considered as neighbors with a degree k , if $d(\bar{a}_1, \bar{a}_2) = k$. The random generator is a generator of alternatives which: (1) avoids generating known alternatives; and (2) is conditioned by the best-known (with respect to the optimization function) alternative discovered in the previous iterations.

Algorithm 1 presents the implementation of the proposed method. The function *check_promising_values* runs the SM on a set of promising alternatives. This set contains all alternatives that have been previously generated as neighbors to a specific best alternative, but have not been evaluated with the DM because the acquisition function has selected other alternatives. This enables one final check of the most promising solutions which may have been missed because of an earlier bad prediction of the SM .

Algorithm 1:

```

Input: Decision model DM, current alternative CA,
Output: best_alternatives
# parameters and initialization
max_e = 150 # maximum number of epochs
n_candidates = 10 # candidates per iteration
objective_jitter = 0.8 # if an alternative is close to the current
                    best (e.g, 75% as good as the current best, the
                    alternative's neighbors should be checked)
random_sample_size = 10000
best_alternatives = []
surrogate_model = new Random_Forest()
promising_alternatives_pool = []
#initial values
candidate_alternatives = generate_random_alternatives(10)
real_objective_values = objective_func(DM, CA, alternatives)
surrogate_model.fit(candidate_alternatives, real_objective_values)
known_alternatives.add(candidate_alternatives,
                        real_objective_values)
best_alternative, best_score = max(candidate_alternatives
                                   ,real_objective_values)
neighboring_alternatives = gen_neighborhood(best_alternative)
while counter < max_e do:
    if size(neighboring_alternatives) > 0:
        alternatives_pool = neighboring_alternatives
    else:
        alternatives_pool = gen_rand_alternatives(best_alternative,
                                                  random_sample_size)
    # get top ranked (e.g., 10) candidates using the acquisition
    function
    candidate_alternatives, candidate_scores =
        perform_acquisition(alternatives_pool, n_candidates)
    #evaluation of candidate alternatives
    real_objective_values = objective_func(DM, CA, alternatives)
    known_alternatives.add(candidate_alternatives,
                           real_objective_values)
    #update current best and promising alternatives
    i=0
    while i < size(candidate_scores) do:
        if best_score * objective_jitter <= candidate_scores[i] do:
            neighboring_alternatives = gen_
                neighbourhood(candidate_alternatives[i])
            promising_alternatives_pool.add(neighboring_alternatives)
            if best_score < candidate_scores[i] do:
                best_alternatives = []
                best_alternatives.add(candidate_alternatives[i])
            if best_score == candidate_scores[i] do:
                best_alternatives.add(candidate_alternatives[i])
            i++
    #update the surrogate model
    surrogate_model.fit(candidate_alternatives, real_objective_values)
    counter++
end
#perform final check of the promising alternatives
best_alternatives =
    check_promising_values(promising_alternatives_pool, best_al
        ternatives)
return best_alternatives

```

4 EXPERIMENTS

4.1 Experimental Setup

The method was evaluated with the 42 decision models described in Section 2. For each decision model, nine different randomly sampled starting alternatives (current alternatives \bar{c}) were sampled. Three of those alternatives were with a final attribute value low, three with a final attribute value medium, and three with a final attribute value high. The desirable outcome was also

varied, i.e., from low to medium, from low to high, from high to medium, and from high to low. This experimental setup resulted in 756 different experimental runs. Each experiment was running for a minimum of 100 epochs, a maximum of 150 epochs, and 50 epochs without improvement. The method and the experiments were implemented in Python, and are available online¹.

4.2 Experimental Results

The average experiment duration for the models with depth 3 was less than 5 min. For the models with depth 4, the duration increased for 3 min and for the models with depth 5 the duration increased for additional 3 min. This indicates that the relation between the computational time and the model depth is linear.

The final output of the algorithm is a set of thousands of different alternatives. However, from a user perspective, only one or just a few alternatives should be enough. Figure 1 presents the number of epochs required to generate the first alternative for the most complex models (depth 5). From the figure it can be seen that on average, the first alternatives are generated in the first 10 epochs. For the less complex models, the number of required epochs was less than 5.

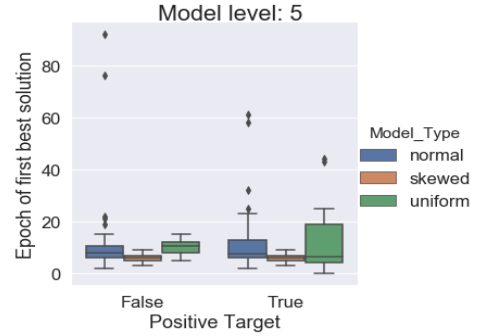


Figure 1: Number of epochs required to generate the first alternative in the final set of alternatives.

In each epoch, the algorithm selects the top 10 alternatives with respect to the optimization score. The higher the score, the better the alternatives are. The selected alternatives depend on the acquisition function, which in turn depends the predictions of the surrogate model. Figure 2 present the average optimization score in each epoch for the most complex models (depth 5). For a comparison, the average optimization score of 10 randomly sampled alternatives at each epoch is also presented (dashed line). From the figure it can be seen that the optimization score of the random samples is significantly lower than the optimization score of the samples selected using the proposed algorithm.

Finally, the presented algorithm is stochastic and the optimality of the solution cannot be guaranteed. One metric that presents the quality of the solutions is the number of attribute changes required to achieve the final solution starting from the current state of the current alternative. Figure 3 presents that metric, which is the same as the distance defined in Section 3.1. From the figure it can be seen that in the majority of the cases, the final solution can be reached with less than 5 attribute changes. Exception of this are the decision models that have a depth 5 and uniform weights' distribution.

¹ [Repository link.](#)

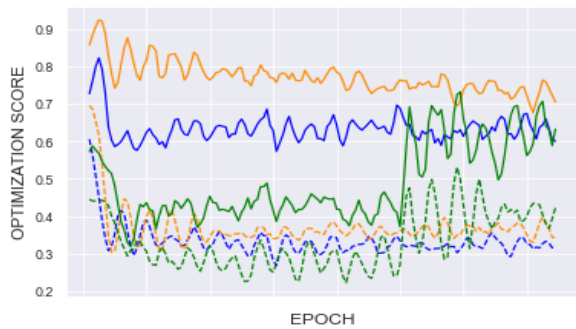


Figure 2: Average optimization score for the decision models with depth 5. Full line - alternatives generated by the surrogate model. Dashed line - random alternatives. The type of attribute weights is color-coded (blue-normal, orange-skewed, green-uniform).

This is because these models have a larger number of input attributes and the uniform distribution requires many attributes to be changed in order for that change to be prolonged to the aggregate attribute. On the other hand, the models with normal and skewed weights' distribution require smaller number of attribute changes for that change to be propagated to the aggregate attributes.

5 DISCUSSION AND CONCLUSION

We presented a novel method for generating alternatives for multi-attribute DEX decision models based on Bayesian optimization. The main goal of the method was to generate alternatives that require the smallest change to the current alternative to obtain a desirable outcome. The method was extensively evaluated on 42 different DEX decision models. The models were with a variable complexity (e.g., variable depth and variable attribute's weight distribution). The method's behavior was analyzed with respect to several characteristics: computing time, time to first appropriate alternative, number of generated (appropriate) alternatives, and number of attribute changes required to reach the generated alternatives.

The experimental results confirmed that the method is suitable for the task i.e., it generates at least one appropriate alternative in less than a minute, even for the most complex decision models. In the majority of the cases, the computing time was lower than that. The discovery of the alternatives was equally distributed throughout the overall runtime. Exception of this is the final check performed by the algorithm (see *check_promising_values* in Algorithm 1), which generates the majority of the alternatives for the more complex models (depth 4 and depth 5). The quality of the alternatives was also appropriate as in the majority of the cases, the generated alternatives could be reached by less than 5 attribute changes. Finally, the relation between the decision-model's depth and the computing time was linear and not exponential, which implies that the method is scalable.

The method implementation considers ordinal attribute values. However, there is possibility for considering other types of distance measures that would work in nominal settings (e.g., Levenshtein distance).

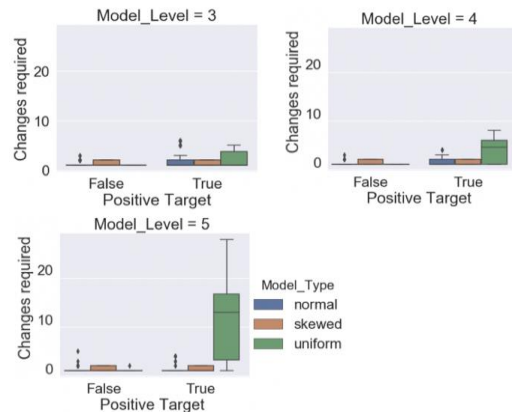


Figure 3: Boxplots for the number of changes required to switch from the starting alternative to the best alternative.

Regarding the future work, the proposed method is stochastic and the optimality of the final solution cannot be guaranteed. In order to do that, the method needs to be validated additionally. Promising options include comparison of the proposed method with deterministic methods and methods that utilize internal rules by which the decision models operate.

REFERENCES

- [1] Power, D.J. Decision Support Systems: Concepts and Resources for Managers. Quorum Books, Westport, 2002.
- [2] Turban, E., Aronson, J. and Liang, T.-P. Decision Support Systems and Intelligent Systems, Prentice Hall, Upper Saddle River, 7th Edition, 2005.
- [3] Mallach, E.G. Decision Support and Data Warehouse Systems. Irwin, Burr Ridge, 2000.
- [4] Sadok, W., Angevin, F., Bergez, J.-E., Bockstaller, C., Colomb, B., Guichard, L., Reau, R., Messeau, A. and Doré, T. MASC: a qualitative multi-attribute decision model for ex-ante assessment of the sustainability of cropping systems. *Agron. Sustain. Dev.* 29, 447–461, 2009.
- [5] Munda, G. Multiple criteria decision analysis and sustainable development. In: *Multiple Criteria Decision Analysis: State of the Art Surveys*, Springer-Verlag, New York, 2005.
- [6] Bohanec, M. and Rajkovič, V. DEX: An Expert System Shell for Decision Support. *Sistemica* 1(1), 145–157, 1990.
- [7] Bohanec, M. and Rajkovič, V. Multi-attribute decision modeling: Industrial applications of DEX. *Informatica* 23, 487–491, 1999.
- [8] Bohanec, M. DEXi: Program for Multi-Attribute Decision Making User's Manual." Ljubljana, Slovenia: Institut Jozef Stefan, 2008.
- [9] Bohanec, M., Zupan, B. and Rajkovič, V. Applications of qualitative multi-attribute decision models in health care, *International Journal of Medical Informatics* 58-59, 191-205, 2000.
- [10] Bohanec, M., Cortet, J., Griffiths, et al. A qualitative multi-attribute model for assessing the impact of cropping systems on soil quality. *Pedobiologia* 51, 239–250, 2007.
- [11] Bohanec, M., Messéan, A., Scatista, S. et al. A qualitative multi-attribute model for economic and ecological assessment of genetically modified crops. *Ecol. Model.* 215, 247–261, 2008.
- [12] Coquil, X., Fiorelli, J.L., Mignolet, C., et al. Evaluation multicritère de la durabilité agr environnementale de systèmes de polyculture élevage laitiers biologiques. *Innov. Agron.* 4, 239–247, 2009.
- [13] Bohanec, M., Cestnik, B., Rajkovič, V. Qualitative multi-attribute modeling and its application in housing. *Journal of Decision Systems* 10, pp. 175-193, 2001.
- [14] Debeljak, M., Trajanov, A., Kuzmanovski, V. et al. A field-scale decision support system for assessment and management of soil functions. *Frontiers in Environmental Science*, 7, p.115, 2019.
- [15] Bergez, J.-E. Using a genetic algorithm to define worst-best and best-worst options of a DEXi-type model: Application to the MASC model of cropping-system sustainability. *Computers and electronics in agriculture* 90: 93-98, 2013.
- [16] Kuzmanovski, V., Trajanov, A., Dzeroski, S., et al., M. Cascading constructive heuristic for optimization problems over hierarchically decomposed qualitative decision space. *Omega*, submitted September, 2020.
- [17] Rasmussen C. E. and Williams C. K.I. *Gaussian Processes for Machine Learning*, MIT Press 2006.
- [18] Frank, H., Hoos, H. H., and Leyton-Brown, K. Sequential model-based optimization for general algorithm configuration (extended version). Technical Report TR-2010–10, University of British Columbia, Computer Science, Tech. Rep. 2010.
- [19] Lizotte F. Practical Bayesian Optimization. PhD thesis, University of Alberta, Edmonton, Alberta, Canada, 2008.

Detekcija napak na industrijskih izdelkih

Defect Detection on Industrial Products

David Golob
Institut Jožef Stefan
Ljubljana, Slovenia
david.golob@ijs.si

Primož Kocuvan
Institut Jožef Stefan
Ljubljana, Slovenia
primož.kocuvan@ijs.si

Jože Ravničan
UNIOR Kovaška industrija d.d.
Zreče, Slovenia
joze.ravnican@unior.com

Janko Petrovčič
Institut Jožef Stefan
Ljubljana, Slovenia
janko.petrovcic@ijs.si

Jani Bizjak
Institut Jožef Stefan
Ljubljana, Slovenia
jani.bizjak@ijs.si

Matjaž Gams
Institut Jožef Stefan
Ljubljana, Slovenia
matjaz.gams@ijs.si

Stefan Kalabakov
Institut Jožef Stefan
Ljubljana, Slovenia
stefan.kalabakov@ijs.si

Gregor Dolanc
Institut Jožef Stefan
Ljubljana, Slovenia
gregor.dolanc@ijs.si

POVZETEK

V članku predstavimo različne metode za detekcijo napak na industrijskih odkovkih. Raziskava je bila narejena v okviru projekta ROBKNCEL. Napake, ki jih želimo zaznati, so manjši udarci ter poškodbe na struženi površini. V začetnih poskusih smo uporabili metode računalniškega vida ter metode zaznavanja napak s tresljaji. Začetni rezultati niso zadovoljivi, vendar nekatere metode kažejo vzpodbudne rezultate, ki bi se jih dalo izboljšati z večjim naborom podatkov.

KLJUČNE BESEDE

Detekcija napak, računalniški vid, tresljaji, industrijski izdelki

ABSTRACT

In this paper different methods for error detection on industrial forks are presented. Part of the research was done for project ROBKNCEL. The types of errors that are detected are mostly scratches and dents on smooth metal surfaces. First a computer vision approach is used and then method for detecting errors from vibrations is discussed. Initial results are not encouraging, but could possibly be improved with larger dataset for training.

KEYWORDS

Error detection, computer vision, vibrations, industrial products

1 UVOD

V zadnjem času so z napredkom strojnega učenja ter umetne inteligence napredovali tudi procesi kontrole kakovosti v industriji. Namen naše raziskave je razviti algoritem za

zaznavanje napak na industrijskih izdelkih/odkovkih za podjetje Unior d.d. Raziskave so bile narejene v okviru projekta ROBKNCEL ([1]), ki ga sofinancira Republika Slovenija iz Evropskega sklada za regionalni razvoj. Klasični pristopi, ki so uporabljeni za detekcijo napak na industrijskih objektih, temeljijo na računalniškem vidu ([2], [3], [4], [5]). V naši raziskavi uporabimo dva pristopa računalniškega vida, in sicer, detekcijo objektov (angl. »object detection«) ter segmentacijo slike (angl. »image segmentation«). Prav tako smo poskusili zaznati napake s tresljaji izdelkov. Glede na inicialne eksperimente, ki niso dali optimalnih rezultatov, se v prihodnje usmerjamo na poskuse strojnega učenja z večjim naborom podatkov ter drugimi, konkretno laserskim čitalnikom, ki se trenutno kaže kot najbolj perspektivna možnost. Raziskave so zanimive predvsem zato, ker so pokazale določene težave v uporabi metod strojne inteligence pri delu z industrijskimi produkti.

2 PRISTOP RAČUNALNIŠKEGA VIDA

V tem pristopu se napake na izdelkih zaznavajo iz navadnih slik. Podani so primeri brezhibnih izdelkov in primeri z napakami, tipično poškodbami na struženi površini. Algoritmi, ki zaznavajo napake, temeljijo na pod-področju strojnega učenja, to je globokega učenja. V zadnjih nekaj letih je področje globokega učenja doseglo izjemne rezultate na področju računalniškega vida, kot npr. detekcija objektov, segmentacija slik ter klasifikacija slik. Pomanjkljivost globokega učenja je, da zahteva velik nabor učnih podatkov. V naših poskusih smo, kot rečeno, uporabili dva (pod) pristopa, to sta, detekcija objektov (angl. »object detection«) ter segmentacija slike (angl. »image segmentation«). Nekaj primerov detekcije napak iz industrijskih

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

Information Society 2020, 5–9 October 2020, Ljubljana, Slovenia
© 2020 Copyright held by the owner/author(s).

izdelkih z uporabo računalniškega vida je opisanih v [2], [3], [4] ter [5].

2.1 Detekcija objektov

V pristopu detekcije objektov tipično skušamo poiskati izbrani objekt (to je lahko npr. avto, pešec, kolo, prometni znak itd.). V našem problemu je izbrani objekt napaka na industrijskem odkovku. Za ta pristop smo imeli na razpolago 9 izdelkov, iz katerih smo naredili nabor 46 slik.

Nabor slik smo nato ločili na učno in testno množico. Delitev je narejena tako, da se isti izdelek ne pojavi v različnih množicah. Na vsaki sliki v učni množici je bilo potrebno ročno označiti napako/napake s pravokotniki. Ko imamo označene slike, jih lahko uporabimo za učenje globoke nevronske mreže, ki je sposobna prepoznavanja objektov (napak) v slikah.

Nevronska mreža je na začetku sestavljena iz več t.i. konvolucijskih slojev (angl. »convolution layers«), na koncu pa imamo par polno povezanih slojev (angl. »fully connected layers«). Konvolucijski sloji so sposobni kreiranja uporabnih značilk (kot npr. razni robovi in oblike na sliki), ki so nato uporabljene v polno povezanih slojih (glej sliko 1 za primer). V primeru detekcije objektov nevronska mreža v prvem delu odkrije t.i. regije zanimanja (angl. »regions of interest«) na sliki, le te regije so v obliki pravokotnikov. Vsaka regija zanimanja je nato vhodni podatek v drugi del nevronske mreže, katere naloga je klasifikacija dane regije (glej sliko 2). V našem primeru smo uporabili že v naprej zgrajeno in naučeno nevronska mrežo, ki smo jo nato »naučili« prepoznavati naše objekte (napake). Nevronska mreža, ki smo jo uporabili, se imenuje »Faster RCNN inception« in je bila naučena na podatkovni množici imenovani »COCO« [6]. Ta nevronska mreža je prosto dostopna ter podprta s strani Python knjižnice *Tensorflow* [7].

Ko imamo naučeno nevronska mrežo, klasificiramo določeno sliko kot »napako«, v primeru da mreža zazna napako z več kot 40% verjetnostjo (glej sliko 3 za primer). V tabeli 1 in tabeli 2 lahko vidimo rezultate mreže na učni množici oziroma na testni množici.

Tabela 1: Učna množica: 27 slik, 26 z napako, 1 brez.

Točnost: 81%, priklic: 81%, natančnost: 100%.

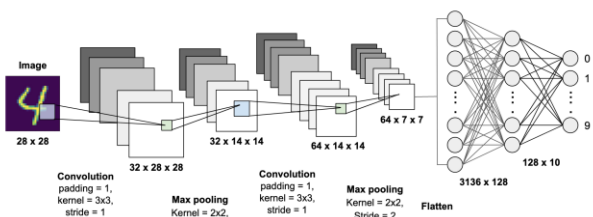
TP	FP	TN	FN
21	0	1	5

Tabela 2: Testna množica: 19 slik, 18 z napako, 1 brez.

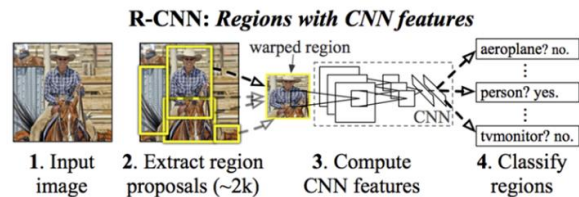
Točnost: 10%, priklic: 5%, natančnost: 100%

TP	FP	TN	FN
1	0	1	17

Opazimo, da na učni množici dobimo zadovoljivo natančnost, vendar model ni sposoben generalizacije, kar se vidi v slabih

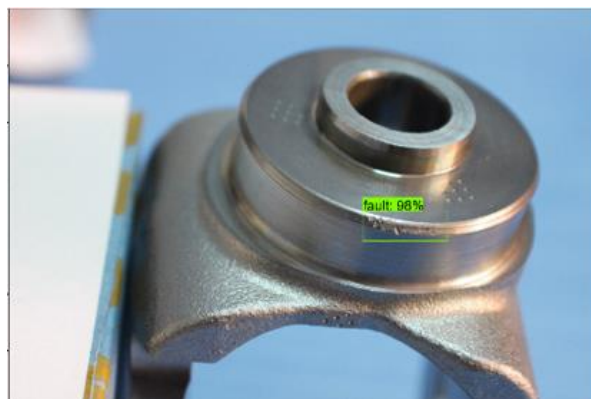


Slika 1: Globoka nevronska mreža s konvolucijami, vir: [8]



Slika 2: Nevronska mreža za prepoznavanje objektov, vir: [9]

rezultatih na testni množici. Za boljše rezultate bi očitno potrebovali več slik in več različnih napak.



Slika 3: Detekcija napak s prepoznavanjem objektov

2.2 Segmentacija slike

V segmentaciji slike klasificiramo vsako slikovno točko v določen razred (glej sliko 4 za primer). V našem primeru imamo samo dva razreda, to sta, »napaka« in »ni-napake«. Tudi v tem pristopu uporabimo (globoke) nevronske mreže za segmentacijo in klasifikacijo.

Za arhitekturo nevronske mreže smo uporabili arhitekturo, ki je bila uporabljena za podoben problem (glej [5] za podrobnosti). Arhitektura je vidna sliki 5. Nevronska mreža je sestavljena iz dveh delov, in sicer, segmentacijskega dela ter klasifikacijskega dela. Vhodni podatek v segmentacijski del je črno-bela slika objekta, klasifikacijski del pa ima dva vhodna podatka (tenzorja) in sicer gre za dva tenzorja iz segmentacijske mreže. Prvi tenzor je segmentacija (pomanjšane) slike objekta, (na sliki 5 je označen kot »segmentation output«) to je tenzor debeline 1, kjer vsak element (ki se ga lahko predstavlja kot slikovno točko) predstavlja verjetnost napake. Drugi tenzor pa je predzadnji tenzor v segmentacijski mreži.

Izhodni tenzor za klasifikacijsko nevronska mrežo je verjetnost, ali slika vsebuje izdelek z napako, za segmentacijsko nevronska mrežo pa je segmentacija pomanjšane slike objekta.

Segmentacijski del se uči ločeno od klasifikacijskega. In sicer, se uči iz ročno označenih slik segmentacije. Klasifikacijski del pa se uči iz binarnih oznak (1 pomeni, da ima objekt napako in 0 pomeni, da slika nima napake).

V tem pristopu razdelimo podatke na učno, validacijsko ter testno množico (kjer noben izdelek ne more biti v dveh množicah). Nato vsako slikovno točko v sliki označimo, kot napako ali ni-napake. To naredimo za vsako sliko v učni in validacijski množici.

Nevronska mreža nam poda segmentacijo slike ter klasifikacijo slike. Primer izhoda nevrnske mreže za segmentacijo je prikazan na sliki 6.

Na validacijski množici smo določili število epoh za učenje mreže in sicer smo za segmentacijsko mrežo uporabili 2900 epoh in za klasifikacijsko nevrnsko mrežo 200 epoh. Za treniranje mreže je bil uporabljen gradientni spust (angl. Gradient Descent) algoritem s parametrom hitrost učenja (angl. »learning rate«) 10^{-3} . Posamezni rezultati so zbrani v tabelah 3,4 in 5.

Tabela 3: Učna množica: 43 slik, 29 z napako, 14 brez napake. Točnost:100%, priklíc: 100%, natančnost: 100%

TP	FP	TN	FN
29	0	14	0

Tabela 4: Validacijska množica: 25 slik, 21 z napako, 4 brez napake. Točnost: 64%, priklíc: 66,7%, natančnost: 87,5%.

TP	FP	TN	FN
14	2	2	7

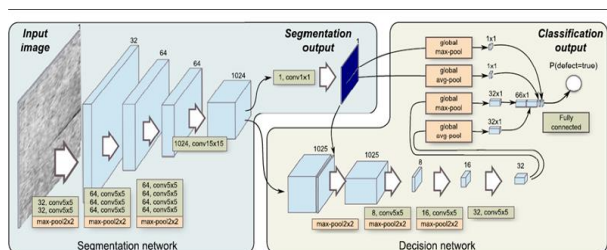
Tabela 5: Testna množica: 28 slik, 21 slik z napako, 7 brez napake. Točnost: 71,4%, priklíc: 81%, natančnost: 81%

TP	FP	TN	FN
17	4	3	4

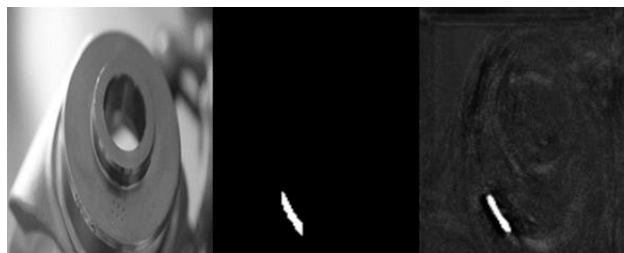
Vidimo, da se je nevrnska mreža sposobna naučiti s 100% točnostjo, vendar ima, podobno kot prejšnji pristop, problem z generalizacijo.



Slika 4: Primer segmentacije slike, vir: [10]



Slika 5: Arhitektura



Slika 6: Primer segmentacije slike. Levo: original, sredina: ročna segmentacija, desno: modelska segmentacija.

3 PRISTOP S TRESLJAJI

Eden izmed "alternativnih", vendar potencialno obetavnih pristopov je analiza na osnovi oscilatornega vzbujanja pomika. Eksperiment je potekal v laboratoriju odseka E2 na IJS. Pozitiv izdelka (dejanski odkovek) smo postavili v negativ (stojalo za odkovke – glej sliko 7) ter generirali oscilatorni pomik negativa (stojala) s pomočjo generatorja vibracij. Zanimalo nas je, ali bi utegnile poškodbe izdelka na naležni površini s stojalom (negativom) kakorkoli vplivati na sklopitev med izdelkom in stojalom. V ta namen smo opazovali dva signala: vzbujevalni signal pomika stojala in izmerjeni signal pomika izdelka ter opazovali odnos med obema. Za vzbujanje pomika negativa (stojala) smo uporabili sinusni vzbujevalni signal. Meritve pomika izdelka smo opravili z laserskim merilnikom razdalje z visoko natančnostjo. Merilnik kontinuirano meri razdaljo do izdelka, ter nato z numeričnim odvajanjem izračuna hitrost, ki je izhodni signal. Za osnovni preizkus smiselnosti metode smo na enem od izdelkov simulirali napako tako, da smo na naležno površino prilepili droben kos izolacijskega traku. Izkazalo se je, da le-ta bistveno vpliva na sklop izdelek-negativ in to nam je dalo upanje, da bi utegnile tudi poškodbe naležne površine izdelka vplivati na sklopitev in s tem na relacijo med pomikom negativa in izdelka.

Posnetki meritve izhodnega signala so dolgi 10s. Meritve smo opravili pod 4 različnimi nastavitvami vhodnega signala, in sicer:

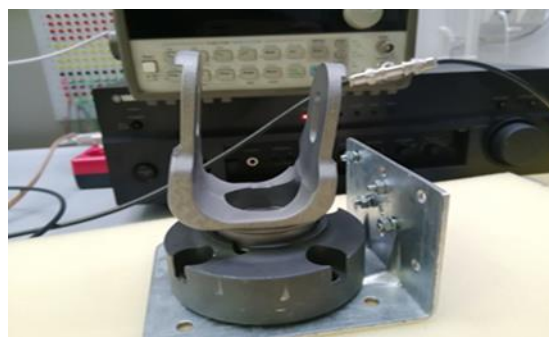
- Nastavitev 1: Amplituda: 0,389 Vpp frekvenca: 50Hz
- Nastavitev 2: Amplituda: 0,389 Vpp; frekvenca: 60Hz
- Nastavitev 3: Amplituda: 0,2026 Vpp; frekvenca: 60Hz
- Nastavitev 4: Amplituda: 0,2026 Vpp; frekvenca 50Hz

Nastavitve so bile izbrane na podlagi izhodnega signala, izkaže se, da za višje amplitude izhodni signal postane šumen.

Za ta pristop imeli na voljo 24 izdelkov.

Preizkusili smo sledeče možne pristope detekcije napak iz signalov:

- Ekspertno izbrane značilke ter uporaba klasičnih metod strojnega učenja.
- Računalniško generirane značilke ter uporaba 2-slojne nevrnske mreže



Slika 7: Meritev vibracij

3.1 Ekspertno izbrane značilke ter uporaba klasičnih metod strojnega učenja

V tem pristopu so značilke, uporabljane v algoritmih strojnega učenja, izbrane na podlagi dobrih izkušenj. Značilke, ki so bile izbrane, so se namreč izkazale kot dobre v drugi aplikaciji strojnega učenja. Izbranih značilk je 22 in uporabljajo osnovne značilke signala iz časovnega ter frekvenčnega spektra, npr. 3 najvišji vrhovi spektralne gostote ter njihove frekvence, energija spektralne gostote, itd.

Vsak posnetek odkovka je razdeljen na 10 kosov, kjer je vsak kos 1s dolg posnetek. Za vsak kos se nato izračuna ekspertno izbrane značilke. Tako za vsak vzorec dobimo 10 podatkovnih točk z 22 značilkami.

Uporabljen model je sestavljen iz dveh modelov. In sicer iz osnovnega ter končnega modela. Osnovni model za vsako podatkovno točko izračuna verjetnost, da ta točka pripada produktu z napako. Ker imamo za vsak produkt 10 podatkovnih točk, dobimo z osnovnim modelom 10 verjetnosti za vsak produkt. Končni model potem klasificira produkt v »odkovek z napako« ali »odkovek brez napake«. Vhodni podatek v končni model je 10 verjetnosti, dobljenih iz osnovnega modela.

Preizkusili smo več možnih algoritmov, in sicer algoritem podpornih vektorjev (angl. »support vector classifier«), algoritem naključnih gozdov, logistično regresijo, algoritem »AdaBoost« ter algoritem »XGBoost«. Te algoritme smo preizkušali tako za osnovni kot končni model.

V prvem poskusu, so bili podatki razdeljeni na učno ter testno množico. Na učni množici smo z 8 delnim prečnim preverjanjem izbrali optimalne parametre za osnovni ter končni model. Nato smo celoten model testirali na testni množici.

Uporabljena je bila nastavitvev 2 vhodnega signala, kjer je bila amplituda 0,389 Vpp s frekvenco 60 Hz. Rezultati so zbrani v tabelah 6 in 7.

Osnovni model: XGBoost

Končni model: Naključni gozdovi

Tabela 6: Učna množica: 19 produktov: 12 z napako, 7 brez napake. Točnost: 100%, priklic: 100%, natančnost: 100%.

TP	FP	TN	FN
7	0	12	0

Tabela 7: Testna množica: 5 produktov: 2 z napako, 3 brez napake. Točnost: 100%, priklic: 100%, natančnost: 100%.

TP	FP	TN	FN
3	0	2	0

Da se izognemo naključnemu dobremu rezultatu na testni množici, uporabimo še drug poskus. In sicer, uporabimo metodo prečnega preverjanja za določanje učne in testne množice. Konkretno uporabimo 5-delno prečno preverjanje, kjer so podatki razdeljeni na 5 delov. Naš postopek ima 5 iteracij, na vsaki iteraciji je en del podatkov izbran kot testna množica, ostali štirje deli pa so izbrani kot učna množica. Na vsaki iteraciji na učni množici z 8 delnim prečnim preverjanjem izberemo optimalne parametre in naučimo model na učni množici, nato pa ocenimo model na testni množici. Ker uporabljamo 5 delov, dobimo 5 ocen točnosti, priklica ter natančnosti, iz katerih nato izračunamo povprečje. (uporabljena je bila nastavitvev 2

vhodnega signala, kjer je bila amplituda 0,389 Vpp s frekvenco 60 Hz). Najboljše testne rezultate so v tabeli 8.

Tabela 8: Osnovni model: logistična regresija. Končni model: AdaBoost

Točnost	Priklic	Natančnost	F1
68 %	85 %	76 %	73 %

3.2 Računalniško generirane značilke ter uporaba 2-slojne nevronske mreže

Za avtomatsko generacijo značilk smo uporabili za to namenjeno knjižnico. Pri nastavljenem parametru FDR (False Discovery Rate) na privzeto vrednost, ki je 0,05 po statističnem testu, nismo dobili nobene značilke, ki bi bila relevantna za klasifikacijo. Ker knjižnica uporablja statistično analizo za ocenjevanje relevantnosti značilk, torej ni nujno, da niso pomembne pri strojnem učenju, zato smo dvignili prag FDR na začetku na 0,5 in nato še na 0,99. Pri tem smo pri vrednosti 0,5 FDR dobili le eno značilko. Ta je 50. Fourierjev koeficient oziroma pri nastavitvi 2 in 3 smo dobili 60. Fourierjev koeficient. Slednja vrednost je seveda osnovni harmonik vzbujalnega signala. Pri nekaterih nastavitvah in pri večji vrednosti FDR smo dobili nekatere Fourierjeve koeficiente v okolici 50. in 60. koeficienta, kar je smiselno, ker je odziv odkovka različen glede na poškodbo. Zaradi tega smo sklenili, da izračunamo Fourierjeve koeficiente v okolici 50. in 60. in jih uporabimo za klasifikacijo. Hevristično smo določili, da izračunamo prvih 256 koeficientov. S tem smo zajeli vse koeficiente v okolici 50. in 60. Izračun prevelikega števila koeficientov pomeni, da lahko porabimo vse vire, ki so na voljo nevronske mreži, prav tako pa uradni viri [11] v tem primeru navajajo 28 x 28 točk oziroma vhodnih nevronov.

Nevronska mreža je sestavljena iz vhodne plasti, ki ima 256 nevronov, nato sledita dve skriti plasti, prva z 16 nevroni, ter druga z 8. Zadnja izhodna plast je sestavljena iz 2 nevronov, ta predstavljata poškodovan ali nepoškodovan odkovek. Takšne nastavitve smo dobili od večkratnega testiranja modela (optimizacija hiperparametrov). Za razliko od prejšnjega pristopa smo uporabili celoten 10-sekunden posnetek za izračun koeficientov.

Kot v predhodnem primeru smo na začetku uporabili optimizacijo hiperparametrov na učni množici. To pomeni, da smo z izbranimi parametri, ki so dosegli najvišjo točnost pri modelu nevronske mreže uporabili za učenje modela. Vseh 24 učnih primerov smo razdelili na učno (19 primerov) in testno (5 primerov). Uporabili smo 5-delno prečno preverjanje kot v prejšnjem primeru. Ker dobimo 5 vrednosti posameznih metrik, na koncu izračunamo povprečje. Rezultati so zbrani v tabeli 9.

Tabela 9: Točnost priklic in natančnost brez F1 metrike

Točnost	Priklic	Natančnost
48 %	42 %	91 %

4 ZAKLJUČEK

V tem prispevku so opisani pristopi ter modeli za detekcijo napak na industrijskih izdelkih - odkovkih.

Rezultati za detekcijo napak z uporabo računalniškega vida in segmentacije slike so se izkazali kot nezadovoljivi za praktično uporabo, kjer se zahtevata visoka točnost in priklic. Rezultati z uporabo računalniškega vida in detekcije objektov so nezadovoljivi najbrž zato, ker so napake na kovini podobne temnim lisam na kovini, ki jih je polno na odkovkih.

Rezultati za detekcijo napak z uporabo tresljajev so vzpodbudni, ampak nezadovoljivi.

Glavni razlog za slabše rezultate je pomanjkanje podatkov ter zajem podatkov v nekontroliranem okolju. Menimo, da ko bo na voljo več podatkov, se bodo rezultati izboljšali.

5 BIBLIOGRAFIJA

- [1] „ROBKONCEL,“ SMM, January 2019. [Elektronski]. Available: http://www.smm.si/?post_id=4682. [Poskus dostopa 30 January 2020].
- [2] M. El-Agamy, M. A. Awad in H. A. Sonbol, „Automated inspection of surface defects using machine vision,“ v *17th Int. AMME Conference*, Cairo, 2016.
- [3] C. Ming , B.-C. Chen , L. G. Jacque in C. Ming-Fu, „Development of an optical inspection platform for surface defect detection in touch panel glass,“ *International Journal of Optomechatronics*, Izv. 10, št. 2, pp. 63-72, 2016.
- [4] X. Sun, J. Gu, S. Tang in J. Li, „Research Progress of Visual Inspection Technology of Steel Products—A Review,“ *Applied sciences*, Izv. 8, št. 11, 2018.
- [5] D. Tabernik, Š. Samo , J. Skvarč in D. Skočaj, „Segmentation-based deep-learning approach for surface-defect detection,“ *Journal of Intelligent Manufacturing*, 2019.
- [6] [Elektronski]. Available: <http://cocodataset.org/#home>.
- [7] Tensorflow, „Tensorflow home page,“ [Elektronski]. Available: <https://www.tensorflow.org/>. [Poskus dostopa 30 January 2020].
- [8] [Elektronski]. Available: <https://towardsdatascience.com/mnist-handwritten-digits-classification-using-a-convolutional-neural-network-cnn-af5fafbc35e9>.
- [9] U. Farooq, 15 February 2018. [Elektronski]. Available: https://medium.com/@umerfarooq_26378/from-r-cnn-to-mask-r-cnn-d6367b196cfd.
- [10] J. Jordan, „Jeremy Jordan,“ 30 March 2018. [Elektronski]. Available: <https://www.jeremyjordan.me/evaluating-image-segmentation-models/>.
- [11] [Elektronski]. Available: <https://www.tensorflow.org/tutorials/keras/classification>. [Poskus dostopa 2019].
- [12] M. Gjoreski, S. Kalabakov, M. Luštrek in H. Gjoreski, „Cross-dataset deep transfer learning for activity recognition,“ v *Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*, 2019.

Data Protection Impact Assessment - an Integral Component of a Successful Research Project From the GDPR Point of View

Gizem Gültekin Várkonyi
University of Szeged
Szeged, Hungary
gizemgv@juris.u-szeged.hu

Anton Gradišek
Jožef Stefan Institute
Ljubljana, Slovenia
anton.gradisek@ijs.si

ABSTRACT

Artificial intelligence and algorithmic decision-making systems help generate new knowledge about diseases which then help better manage it and assist people in clinical treatment needs. The blood of such AI systems is personal data that is both used for training or is already the output of the algorithmic assessments. This work aims guiding the AI researchers to be familiar with the legal rules binding them while processing personal data within their AI-based projects as indicated in the General Data Protection Regulation rules with a specific focus on why and how to conduct a self-Data Protection Impact Assessment. The self-assessment guideline presented throughout the work is an output of the mutual experiences and collaboration between a lawyer and an AI researcher on the topic.

KEYWORDS

data protection, impact assessment, GDPR, artificial intelligence, medical data

1 Introduction

It is possible to look out for artificial intelligence (AI) systems dealing with personal data from two different perspectives. On one hand, it offers great benefits for the users, developers, and researchers, if used correctly. For example, AI-enabled health care technologies could predict the treatment of diseases 75% better, and could reduce the clinical errors 2/3 at the clinics using AI compared to the clinics that do not [1]. On the other hand, the improper handling of personal data can quickly lead to abuse, sharing sensitive information, or other problems (unwanted data disclosure, complex and costly legal procedures, high fines, etc.), therefore it has to be handled with the utmost care. In this paper, we will focus on the legality of medical applications containing personal data that is defined as sensitive data in legal documents, such as the analysis of sensor data to help patients with chronic diseases manage their condition and improve the quality of life, or to help the elderly with independent living by providing safety features and improved communication channels.

Developing an AI-based service for a target population, for example people with diabetes, chronic heart failure, obesity, dementia, skin cancer, etc., typically starts with a research project. One of the key components of such a project is collecting substantial amounts of data in a pilot study, with participants that resemble the target audience for the final service. When planning the pilot study, researchers enter a slippery terrain of dealing with personal data, as the participants are providing their own data for the purpose of the study. For the illustration, we can imagine a project where we collect medical data of three types; general medical data provided by the medical doctor responsible for the participant, lifestyle data collected by either wearable or stationary sensors, and self-reported data that is obtained via questionnaires that the participants fill.

The data provided by the participants fall under the scope of the European Union's General Data Protection Regulation (GDPR) since it refers to identified or identifiable personal issues of them. The GDPR entered into force on the 25th of May 2018 with one of the aims of keeping up with the technological developments challenging efficient protection of personal data [2]. The risk-based approach embedded in the GDPR came along with several safeguards as one of them is the Data Protection Impact Assessment (DPIA). The DPIA can help AI-researchers to comply with the GDPR requirements at an early stage of a new project. It can help reduce the risks arising from the use of AI technologies challenging the efficient protection of fundamental rights and principles [3]. Several policy papers generated by the EU institutions [4] [5] focusing on regulation of AI state that legal compliance is a keyword for gaining user trust and DPIA is one way to reach user trust. However, there is no standard set for conducting a DPIA that could guide the AI-researchers. In this paper, we present some of the key points of conducting the DPIA that could be useful for the AI-researchers.

2 Data Protection Impact Assessment in the GDPR

The term DPIA was not specifically described in the GDPR, however, was referred as it is a process to help managing the risks to the data subjects' (participants of the research project, in this case) rights and freedoms as a result of data processing. In other words, DPIA is a process consisting of several other sub-processes to describe the risks and assess the legality of the system in terms of data protection. These risks could be related to system security, system design, implementation,

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
Information Society 2020, 5–9 October 2020, Ljubljana, Slovenia
© 2020 Copyright held by the owner/author(s).

administration and development on a further run. The aim of the DPIA is to take appropriate safeguards to minimize the risks, if impossible to eliminate all. DPIA is not a simple one-time reporting activity, it is an ongoing process that should be continuously carried out during the lifetime of a project, therefore DPIA should always be monitored and updated [6].

It is the AI-researcher's responsibility to convey a DPIA when the data processing activity is likely to constitute a "high risk" to the rights and freedoms of natural persons (e.g. users of an AI service who both benefit from the service and contribute to it with their data). How to decide whether a certain data processing activity would be resulting in a high risk is not an easy task, but there are several guidelines and list of processing requiring DPIA published by the National Supervisory Authorities [7]. These lists could be the first sources for the data controllers to decide about the necessity of the DPIA for a certain project [8].

Failure to conduct a right DPIA raises a risk for the AI-researchers; they may face several sanctions, especially financial penalties. Apart from that, conducting a right DPIA would be beneficial for the data controllers not only from the legal and the financial point of view. A DPIA could help data controllers to avoid implementing irrelevant solutions from the beginning of the project which may refer to assessing the technical feasibility of the system in parallel with the legal compliance [8]. Therefore, the DPIA could help data controllers to save time and money. It also prevents the companies from losing their reputation (or from the scandals, as such occurred with the Cambridge Analytica, Equifax, Facebook, etc.). Finally, a DPIA document can prove the trustworthiness of the project team before the public, as well as the related authorities, since it is an evidence of the respect towards the right to data protection.

An AI project aiming to collect personal data and evaluate the data with an automated decision-making system with the help of profiling tools such as surveys and hardware equipment must be assessed from the risk point of view. Below, we present a step-by-step guideline on how to conduct a DPIA on AI-based research.

3 Conducting a Data Protection Impact Assessment

In this section, we assume a project aiming at developing a medical software with the help of an algorithm that is going to enable collecting and processing participants' sensitive data based on profiling. Additionally, a large amount of data will be collected for feeding the algorithm, meaning that the participants may lose a degree of control of their data stored and processed by the AI system. Based on these inputs, the project may reveal risks for rights and freedoms of the data subjects involved, if these are not mitigated. Therefore, we need to conduct a DPIA and identify the risk categories with the planned mitigations.

We identified three steps for conducting a successful DPIA in the project: the Data Specific Assessment, the Data Subject Specific Assessment, and the Project Specific Assessment.

The **Data Specific Assessment (DSA)** is the procedure where the data to be used in the AI project should be introduced very specifically in order to comply with the basic rules of the GDPR, mainly, the purpose limitation, transparency, accuracy, data minimization, and consent. It should be kept in mind that one of the requirements to be ensuring a valid consent is identifying the concrete data list, together with the planned processing activities of that data in the frame of a research project. Information serving to identify the persons involved with data processing are the natural elements of the DSA. For example, AI-researchers in the project should identify the data processing purposes specific to the project aims and present the list of purposes in a written form to the participants. The indicated purposes should follow the related data to be processed listed again in a written form, followed by the clear identification of the AI-researchers and other people involving the processing activity.

Next, the **Data Subject Specific Assessment** should follow the procedure where the focus is on explaining all the details about how the AI-researchers will ensure the rights of the participants by protecting their informational self-determination right. The key point in this assessment is to gain trust of the participants as required by law and ethics. One of the key aspects here is to make sure that the participants are introduced by the project team on the ways their data will be used, as well as the possibility for them to request removal of their data if so desired. The project team shall also ensure that the participants have a certain degree of accession to the decisions made by the algorithm about them. Explaining an algorithmic decision relating the participants' personal assessment should be understandable to them since the classification models based on decision trees are easily comprehensible to humans. On the other hand, models that are based on complex multilayer neural networks are essentially black boxes where it is not possible to determine why a particular decision was reached based on easily interpretable rules. Bearing in mind the black box nature of the algorithmic assessment, choosing a model that is firstly understandable and explainable to the AI-researchers is a suggested action in this sense. The social implications of choosing a black box algorithm is an emerging research field. Finally, the project team should ensure that the system offers tools for the participants to keep their data accurate and to block third party access.

The **Project Specific Assessment** is the last part of the DPIA, presenting and explaining the legal basis for data processing, the external project partners involved with data processing activities, and the security measures that will be implemented to safeguard the data processed during the project. As the project likely deals with sensitive medical data, security protocols have to be elaborated, which include proper hierarchy regarding the data access, encryption algorithms, regular security updates, and physical access to the hardware where the data is located.

The final but an ongoing phase of the DPIA is the monitoring phase. Whenever there is a new element embedded in the project, and this element seems to change the balance of the risks that were assessed earlier, the DPIA should be reviewed. This element could be involving a new data type in the algorithm or planning a commercial use of the algorithm. Bearing in mind the fact that machine learning techniques and algorithms are referred

to as entirely new technologies [3] and the growing amount of data together with a variety of hardware would raise risks to persons' right to data protection [9], we suggest the project team to review the DPIA periodically, for instance, every year at least.

4 Conclusion

Data Protection Impact Assessment is an integral part of any research project focusing on development of an AI algorithm with personal data. Such data might be sensitive in nature, such as medical data, to be used for developing an algorithm to detect diseases. Besides it is a legal requirement as provided for by the GDPR, a DPIA is a tool for the AI-researchers to assess the weaknesses in the system that may then risk the protection of fundamental rights of the persons participating in the research project who contribute to the development of the project with their personal data. Since there are few guidelines on how to conduct a DPIA for a research project specific to the topic, this work initiates a step-by-step guideline for the AI-researchers.

The first step considers a Data Specific Assessment that the data and the purposes of the data processing are clearly identified and listed in a written form to be presented to the participants. It is followed by the Data Subject Specific Assessment which focuses on the ways the AI-researchers ensure the protection of the participants' right to data protection in line with the GDPR requirements. Such requirements include providing explanation on the decisions reached as a result of algorithmic assessments. The third step relates to the Project Specific Assessment and this step focuses mostly on the security measures planned to be taken by the project team to mitigate the risks that appeared during the previous two assessments. We would suggest the AI-researchers review the DPIA at least once a year, otherwise revision is required whenever a new element is added to the system ending with a new data processing.

From the planning stage of the project to the annual revisions, the DPIA could help the project team to identify the potential

risks and find mitigation strategies for certain weak points. Last but not least, by conducting the DPIA, the project team fulfills the legal requirements, ensures higher trust of people involved, and avoids unforeseeable problems that might later occur.

ACKNOWLEDGMENTS

This work was supported by the ERA PerMed project BATMAN, which was financed on Slovenian side by the Ministry of education, science, and sport (MIZŠ). An extended version of this paper was submitted to journal *Informatica*.

REFERENCES

- [1] "The AI effect: How artificial intelligence is making health care more human", [Online], study conducted by MIT Technology Review Insights and GE Healthcare, 2019. Accessed from: <https://www.technologyreview.com/hub/ai-effect/> Last accessed: 20 April 2020.
- [2] EDPS (2012). "Opinion of the European Data Protection Supervisor on the data protection reform package", (7 March 2012).
- [3] ICO (2018). Accountability and governance: Data Protection Impact Assessments (DPIAs).
- [4] European Commission (2018). Communication from the Commission to the European Parliament, the European Council, the Council, the European Economic and Social Committee and the Committee of the Regions, Artificial Intelligence for Europe. COM (2018) 237 final.
- [5] European Commission (2018) Communication from the Commission to the European Parliament, the European Council, the Council, the European Economic and Social Committee and the Committee of the Regions, Coordinated Plan on Artificial Intelligence. COM (2018) 795 final.
- [6] Wright, David. (2012). The state of the art in privacy impact assessment. *Computer Law & Security Review*, 28(1), 54–61. <https://doi.org/https://doi.org/10.1016/j.clsr.2011.11.007>
- [7] Hungarian National Authority for Data Protection and Freedom of Information (NAIH), List of Processing Operations Subject to DPIA 35(4) GDPR <https://naih.hu/list-of-processing-operations-subject-to-dpia-35-4-gdpr.html>
- [8] Wright, David. (2011). Should Privacy Impact Assessments Be Mandatory? *Commun. ACM*, 54(8), 121–131. <https://doi.org/10.1145/1978542.1978568>
- [9] Chandra, Sudipta., Ray, Soumya., Goswami, R.T. (2017). Big Data Security: Survey on Frameworks and Algorithms, in 2017 IEEE 7th International Advance Computing Conference (IACC), Hyderabad, pp. 48-54. doi: 10.1109/IACC.2017.0025

Deep Transfer Learning for the Detection of Imperfections on Metallic Surfaces

Stefan Kalabakov
stefan.kalabakov@ijs.si
Jožef Stefan Institute
Mednarodna podiplomska šola
Jožefa Stefana
Ljubljana, Slovenia

Primož Kocuvan
primoz.kocuvan@ijs.si
Jožef Stefan Institute
Ljubljana, Slovenia

Jani Bizjak
jani.bizjak@ijs.si
Jožef Stefan Institute
Mednarodna podiplomska šola
Jožefa Stefana
Ljubljana, Slovenia

Samo Gazvoda
samo.gazvoda@gorenje.com
Gorenje gospodinjski aparati, d.d.

Matjaž Gams
matjaz.gams@ijs.si
Jožef Stefan Institute
Mednarodna podiplomska šola
Jožefa Stefana
Ljubljana, Slovenia

ABSTRACT

In the last decade, consumers' expectations have significantly increased regarding the availability and quality of the products they buy. To this end, manufacturers have focused on streamlining their manufacturing lines by employing intelligent solutions wherever possible. Since the field of quality control remains dependent mainly on specialized workers, interest in incorporating artificial intelligence (AI) advances in this field has dramatically increased. In this paper, we present a short exploration into a computer vision system built to detect imperfections on metallic surfaces. In particular, we leverage deep transfer learning to build a model that can classify small segments of a bigger image while using a tiny dataset for training. In these initial experiments, we show that layers trained on the ImageNet dataset can be used as feature extractors when building a model for a vastly different problem.

KEYWORDS

deep transfer learning, computer vision, quality control

1 INTRODUCTION

Today, products are expected to be available fast, in vast quantities, and with exceptional quality. To this end, manufacturers have started streamlining their manufacturing lines by employing network-connected intelligent machines wherever possible [10]. This has created great interest in incorporating advances in artificial intelligence (AI) in the industry. In recent years industrial adoption of AI is becoming more and more feasible [7], mainly thanks to the significant progress in hardware computational resources.

In spite of this, quality control is one manufacturing process which still remains highly dependent on expert human workers. This dependence, in some instances, makes it slower, more prone to errors, and more expensive. To mitigate this, there has been limited adoption of computer vision systems paired with classical

image processing for detecting imperfections in the manufacturing processes [1]. However, these systems rely heavily on specialized lighting solutions in order to highlight imperfections on the surfaces of objects [6]. The systems are usually expensive and require close proximity to the object which is being investigated in order to provide good detection accuracy. Furthermore, methods which do not use any kind of learning require features which are hand-crafted for each application specifically and require some degree of uniformity in size and shape of the errors which might appear. This problem with hand-crafted features, for us, exists even when using classic machine learning models, as we were not provided with details regarding the size and shape of the errors. To solve this, we opted to use deep learning models, as they automatically extract features based on the training set and have proved to produce state-of-the-art results in many areas [3]. With this in mind, the aim of this paper is to investigate low cost state-of-the-art deep learning methods which work in suboptimal lighting and which automatically extract features which are robust to the shape and size of the errors which appear on metallic surfaces. Finally, since our dataset is extremely small, we leveraged transfer learning in order to use the full potential of deep models.

2 PROBLEM DEFINITION

The ultimate goal of the ROBKNONCEL project is to create a quality control process for the detection of several possible manufacturing errors on both the inside and outside of ovens. In this work, we focus on detecting scratches and dents, i.e., imperfections on the oven faceplates' metallic surface. We perform this quality check in the manufacturing process's final phases, as almost fully assembled ovens get transported on a conveyor belt. In order to produce a method that is least costly to implement, we chose a simple RGB camera as the sensor in this application. The camera is positioned such that it can take a picture that contains the whole metallic surface while not interfering with other quality control processes, thus improving efficiency. Finally, our method is supposed to highlight the areas where dents and scratches are found so that an inspection of the algorithm's work can be done at any time. Figure 1 shows an example image used for the purposes of this paper.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

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

© 2020 Copyright held by the owner/author(s).



Figure 1: An image taken by an RGB camera of the metallic surface of interest.

2.1 Data

Due to the frequency with which these imperfections occur, we did not have a large dataset to include in this study. On the contrary, the number of faceplates we could use to get the necessary number of images was only five. Of those five faceplates, one was without imperfections, and the rest contained a varying number of defects on the metallic surface. Since any deep learning requires a large amount of data and since the number of faceplates is small, using images that portray the whole area of one faceplate as examples to a deep neural network (DNN) would be ineffective. To combat this problem, we took images of the different front panels (five images in total) and segmented them into hundreds of smaller examples, which we use as inputs to fine-tune several models. Additionally, by performing class-invariant transformations on these smaller images, we attempt to diversify the set of examples used to fine-tune the models. The segmentation of images into smaller examples and their augmentation are presented in subsection 3.1 and subsection 3.2, respectively.

3 METHOD

3.1 Segmentation

In order to segment the images, we first created a hand-annotated set of binary images (masks). These masks complement the original set of five images by showing where in them, a scratch or a dent is visible on the metallic surface. In more detail, the masks were produced by having humans mark the exact locations of these imperfections. In the masks, pixels which are part of some imperfection (in the RGB image) are marked with the color white, while all others are represented in black. An image and its corresponding mask are shown on Figure 1 and Figure 2, respectively.



Figure 2: A mask constructed for the image in Figure 1.

The next step in the segmentation process is to divide the image into chunks (windows). We do this by "sliding" a window with a fixed size across the whole image. Each of these windows covers a specific area of the image and will serve as a training or testing instance when fine-tuning the models. Overlap between several windows is allowed in fact, it is encouraged, seeing that some overlap means that we can generate more examples. The size of the window is 200 by 200 pixels and the allowed overlap between windows is 75%.

However, since in this paper's scope, we are only interested in the faceplate's metallic parts, we make sure that none of the windows cover an area that includes the display. In Figure 3 we can see (in green) the windows produced by the segmentation and how none of them overlap with the area of the display.

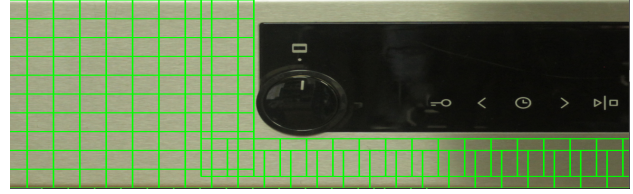


Figure 3: Example of image segmentation.

Finally, since the newly constructed windows will be used to train a deep learning model, we need to assign a label to each one of them. In this application, the labels are "0" and "1". If the label "0" is assigned to a window, it means that the window's area does not include any scratches or dents on the surface. On the other hand, the label "1" means that the area covered by the window includes a scratch or a dent. The labels are assigned to each window by examining the mask. For each window, we take the corresponding area it covers in the mask, and if it includes a certain number of pixels annotated as belonging to an imperfection, then the window is assigned a label "1". Otherwise, it is assigned the label "0". The number of pixels that are used as a threshold for labeling the windows is:

$$threshold = 0.1 \times numPixelsInWindow$$

3.2 Augmentation

Augmentation of images in the data-space has been shown to produce great results when it comes to improving the accuracy of classifiers [5]. Since after segmenting the image, the number of examples (windows) that do not contain an imperfection is largely greater than the number of examples that do, we apply certain transformations to the windows that contain an error, and we save each of those transformed windows as a new example. It is important to emphasize that none of these transformations affect the example's label, meaning that if we apply them to an example containing an error, the transformed example will also contain the same error. The transformations we use are:

- rotation
- change of contrast
- change of brightness
- flipping

After applying these transformations to a single example, 23 new samples are obtained.

3.3 Deep Transfer Learning

For the task of classifying windows based on whether they contain an imperfection or not, we tested four different model architectures. One is a simple Convolutional Neural Network (CNN), and the other three are more complicated architectures that are well established in the world of image recognition.

The simple CNN is used as a baseline for what an end-to-end model can achieve on this dataset. However, since the number of examples is still relatively low, training an end-to-end deep learning model was not expected to yield great results.

On the other hand, the VGG16 [8], InceptionV3 [9] and ResNet101V2 [2] architectures were used to leverage deep transfer learning [4]. To be more specific, all of these networks have been used in the ImageNet competition, and their internal parameters (weights), from that competition, are openly available for use. By using their pretrained convolutional layers as feature extractors and

training our own set of fully connected layers, we can significantly improve our performance and training time. Effectively, we transfer the knowledge stored in their parameters (weights) from the ImageNet dataset to our quality control problem.

To implement this, in every architecture, we disregard the fully connected layers included with these architectures and generate our own (with random weights). We then attach these fully connected layers to the output of the convolutional layers (provided as pretrained on ImageNet) and train only the fully connected layers while freezing the convolutional layers' parameters. The number of fully connected layers we generate is four, and the number of neurons per layer is 512, 256, 128, and 64, respectively.

The implementation and the weights of these models are acquired from the Keras package in TensorFlow.

4 EVALUATION

4.1 Experimental Setup

We evaluated the performance of each model using Leave One Image Out (LOIO) cross-validation. This means that models are trained using examples (windows) from all images but one, and are tested using the instances from the image excluded in the training process. The process is repeated several times, and each time a different image is used to test the models' performance. Since one of the faceplates did not have any errors on its surface, windows from that image were never used to test models, instead they were always used for training. In summary, all the models are evaluated using a 4-fold LOIO cross-validation.

4.2 Evaluation Metric

In this work, we use F1-score with macro averaging as the metric for the evaluation of the models. In particular, we use (macro) F1-score to determine the model's ability to classify segmented windows. The choice to use (macro) F1-score rather than accuracy was made because of the class imbalance in our data. A significant difference between accuracy and (macro) F1-score comes from the fact that accuracy reports a higher value, even in many false positives. For example, a high accuracy score will be reported when a classifier predicts only positive values on a test set containing many positive examples, even though the classifier completely misclassifies the negative instances.

To fully understand the classification results, aside from the F1-score metrics, we also visually represent how the predictions look once all windows have been rearranged in their initial positions. This representation overlays windows in their original places but changes their pixels' value to all white or black based on their predicted values. An example of this representation is shown in the middle image in the triplet on Figure 4. The top image in that same figure changes the pixels' values based on the ground-truth rather than prediction value. Finally, the figure's bottom image represents a color-coded version of the difference between the top two images. Windows in green represent windows which have been predicted as **containing** a fault, when in fact they **do** contain a fault (True Positive - TP). Windows in red represent windows which have been predicted as **not containing** a fault, when in fact they **do** contain a fault (False Negative - FN). And finally, windows in blue, represent windows which have been predicted as **containing** a fault, when in fact they **do not** contain a fault (False Positive - FP).

This view is especially useful for our evaluation since it allows us to filter out wrongly classified windows which surround green

clusters. This dismissal is possible because finding the exact margins of the imperfections is not of great importance in our use case.

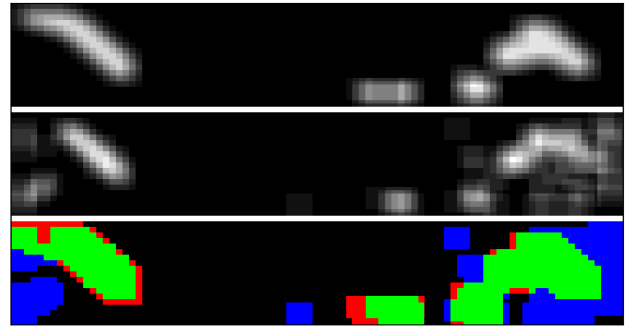


Figure 4: Example of the custom visualisation metric. The top image has the colors of the windows selected based on the groundtruth, while the middle one has them selected based on the predictions of the classifier. The bottom image represents a color-coded version of the difference between the top two images.

5 RESULTS AND DISCUSSION

Table 1 shows the average (macro) F1-scores that each of the models achieved when performing 4-fold LOIO cross-validation.

Architecture	Simple-CNN	VGG16	InceptionV3	ResNet101V2
(Macro) F1-score	-9.8%	-13.0%	58.75%	60.75%

Table 1: Average model F1-score after 4-fold LOIO cross-validation.

In all of our experiments, the Simple-CNN and VGG16 architectures produce very low results. It is our opinion that perhaps, a simple stacking of convolutional layers is not enough for this particular use case, since both networks are unable to learn and instead predict every example as an example with an error. On the other hand, InceptionV3 and ResNet101V2 produce good results in comparison to the other two architectures. A head to head comparison of the per image F1-scores of the two best models can be found in Table 2.

	IMG_8334	IMG_8319	IMG_8327	IMG_8323
InceptionV3	52.48%	64.80%	49.16%	74.04%
ResNet101V2	59.62%	62.02%	51.14%	71.41%

Table 2: Per image F1-scores for the InceptionV3 and ResNet101V2 models.

Although there is only a small difference between the F1-scores of InceptionV3 and ResNet101V2, only 2% as seen on Table 1, there is a large difference in how they predict the same images, as we can see in Figure 5 and Figure 6.

As is clearly visible, ResNet101V2 produces a lot more false positives in comparison to InceptionV3. However, if we consider

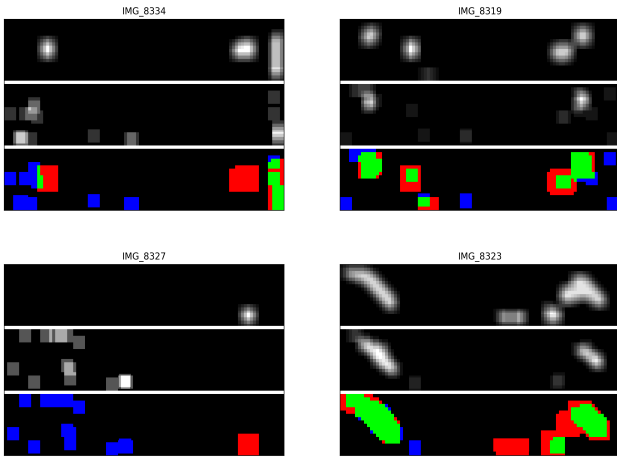


Figure 5: Visual representation of the predictions produced by the InceptionV3 model.

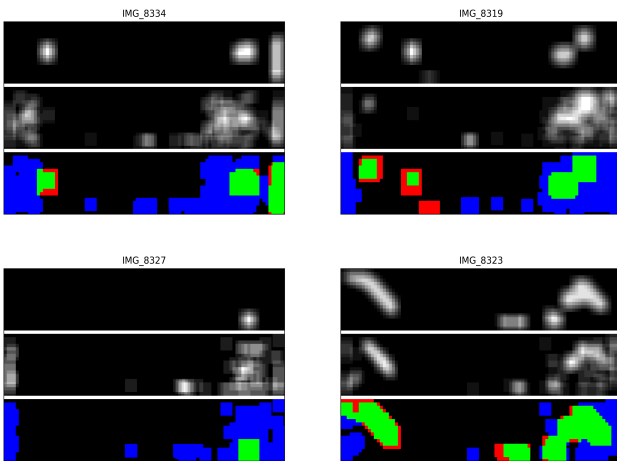


Figure 6: Visual representation of the predictions produced by the ResNet101V2 model.

each cluster of same colored pixels as an error, we can see that ResNet101V2 produces far better results when it comes to the number of true positives and false negatives. So, even though ResNet101V2 produces a lot of false positives, it only manages to miss one error from all four images, whereas InceptionV3 manages to miss four errors. These results can be seen on Table 3. When counting the clusters, we do not consider red clusters surrounding green clusters as a false negative.

	TP	FN
ResNet101V2	11	1
InceptionV3	8	4

Table 3: A sum of the number of true positives and false negative clusters for each of the models across the four test images.

6 CONCLUSION AND FUTURE WORK

In this paper we presented a deep transfer learning approach to quality control in the case where imperfections on metallic

surfaces should be detected. Based on the results it seems that transfer learning is a suitable tool for use when the target dataset is really small, even in the case when the source and target problems are vastly different. Furthermore, it seems like more complex architectures produce better results compared to more traditional ones. When more examples of faceplates with imperfections become available, we plan on exploring the effects of fine tuning some of the convolutional layers in these models rather than freezing all of them during training. Another possible path to take in the future includes using GANs in order to generate realistic looking samples of windows with imperfections and further augmenting our training set. Finally, it is important to note that exploring more appropriate lighting solutions might produce better results.

ACKNOWLEDGMENTS

Part of this research was done under and for ROBKONCEL project. Additionally, this research was partly funded by the Slovene Human Resources Development and Scholarship Fund (Ad futura).

REFERENCES

- [1] Fernando Gayubo, José Luis Gonzalez, Eusebio de la Fuente, Felix Miguel, and Jose R Peran. 2006. On-line machine vision system for detect split defects in sheet-metal forming processes. In *18th International Conference on Pattern Recognition (ICPR'06)*. Volume 1. IEEE, 723–726.
- [2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Identity mappings in deep residual networks. In *European conference on computer vision*. Springer, 630–645.
- [3] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature*, 521, 7553, 436–444.
- [4] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. 2014. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1717–1724.
- [5] Luis Perez and Jason Wang. 2017. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*.
- [6] Franz Pernkopf and Paul O’Leary. 2002. Visual inspection of machined metallic high-precision surfaces. *EURASIP Journal on Advances in Signal Processing*, 2002, 7, 650750.
- [7] Michael Sharp, Ronay Ak, and Thomas Hedberg Jr. 2018. A survey of the advancing use and development of machine learning in smart manufacturing. *Journal of manufacturing systems*, 48, 170–179.
- [8] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- [9] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2818–2826.
- [10] Chris J Turner, Christos Emmanouilidis, T Tomiyama, Ashutosh Tiwari, and Rajkumar Roy. 2019. Intelligent decision support for maintenance: an overview and future trends. *International Journal of Computer Integrated Manufacturing*, 32, 10, 936–959.

Fall Detection and Remote Monitoring of Elderly People Using a Safety Watch

Ivana Kiprijanovska
Department of Intelligent
Systems
Jožef Stefan Institute,
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
ivana.kiprijanovska@ijs.si

Jani Bizjak
Department of Intelligent
Systems
Jožef Stefan Institute,
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
jani.bizjak@ijs.si

Matjaž Gams
Department of Intelligent
Systems
Jožef Stefan Institute,
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
matjaz.gams@ijs.si

ABSTRACT

As seniors age, the risk of unforeseen accidents that affect their well-being increases. Therefore, monitoring the day-to-day routine of elderly people is an important precaution to undertake, especially when they are living alone. Due to the rapid demographic change and aging of the population, the development of remote monitoring systems has become the center of attention for both researchers and industries. In this paper, we present the design of a safety watch integrated in a comprehensive health monitoring system capable of observing the elderly remotely. It integrates low-power hardware architecture and energy-efficient software configuration, which significantly extend the battery autonomy of the device. One of the major modules running on the safety watch is the automatic detection of falls and similar dangerous situations. For that purpose, several machine learning methods were tested, among which the Random Forest method achieved the highest accuracy in detection of falls on data recorded from 17 participants, and was implemented on the actual device.

KEYWORDS

Safety watch, remote monitoring, energy efficiency, fall detection

1 INTRODUCTION

More than 90% of the elderly desire to live in their own homes for as long as they possibly can [1]. However, as seniors age, the risk of unforeseen accidents that affect their well-being increases. For example, the lives of elderly people are very often affected by falls, which lead to not only physical injuries but also psychological consequences that further reduce their independence and decrease the quality of their life [2][3]. The lack of independence causes them to no longer feel comfortable with living alone, forcing them to move into nursing homes. It

puts a burden on the health-care system with over-crowded nursing homes and hospitals, and causes higher health-care expenditures [4]. Therefore, monitoring the day-to-day routine of the elderly who live alone is an important precaution to undertake.

Remote health monitoring systems are essential for enhancing care in a reliable manner and allow the elderly to remain in their home environment rather than in expensive nursing homes [5]. Such systems also allow communication with remote healthcare facilities and caregivers, thus allowing healthcare personnel to keep track of the elderly's overall condition and respond, if necessary, from a distant centralized facility [6]. Due to the rapidly increasing aging population, such technologies have become a subject of interest for both researchers and industries.

One of the first remote monitoring systems presented in the literature are camera-based systems. They are capable of recognizing complex gait activities, but restrict the movement of the user within a specific range. Apart from that, they are complex, expensive and often related to privacy concerns. A recent survey gives an insight to the studies carried out in vision-based patient monitoring [7]. In the last few years, wearable motion sensors have gained in popularity for monitoring human activities in real time. They can monitor and record real-time information about one's physiological condition and motion activities. Wearable sensor-based health monitoring systems may comprise different types of sensors that can be integrated into textile fiber, clothes, and elastic bands or can be directly attached to the human body. One such system is presented in [8], which uses mobile phone as an intermediary to get vital data from various sensors and transmit data to a server for further processing. The main limitation of this system is the fact that the analysis is not performed in the place where the signal is acquired, and there may be a loss of efficiency in the wireless network when physiological signals are sent. Another wearable personal healthcare system is presented in [9]. It employs a number of wearable sensors to continuously collect users' vital signals and uses Bluetooth devices to transmit the data to a mobile phone, which can perform on-site vital data storage and processing. After local data processing, the mobile phone periodically report users' health status to a healthcare centre. Apart from such systems, various wearable commercial products are available on the

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
Information Society 2020, 5–9 October, 2020, Ljubljana, Slovenia
© 2020 Copyright held by the owner/author(s).

market, for example the biometric shirt by Hexoskin, and fitness trackers by Fitbit and Jawbone. However, many current solutions either provide insufficient functionalities at a reasonable price or are advanced but too expensive, too energy demanding or too invasive [10].

The aim of the HomeCare2020 project was to provide a comprehensive solution for a smart healthcare monitoring system, capable of observing the elderly remotely, while eliminating the problems mentioned before. The system aimed to enable the elderly to live home independently until later age and to make them feel safer and more confident in performing everyday tasks and activities. The developed system integrates two interconnected devices: advanced touch-screen care-phone (HomeTab) and a multifunctional safety watch. In this paper, the design of the safety watch is presented.

2 SAFETY WATCH DESIGN

The safety watch is a custom-made wristband device meant to be carried by seniors to provide 24/7 security, inside or outside of the home.

Its core part, from a hardware perspective, is an ARM-based low-power Bluetooth module by Nordic [11]. The priority on choosing the processors and other hardware components was given to how much energy they consume, since a device that requires everyday charging is strongly undesirable, especially for the elderly, who might have problems remembering when or how to charge the device. The safety watch integrates a low-power LSM6DSL system-in-package featuring a 3D digital accelerometer and a 3D digital gyroscope. As well as that, it contains a low-power Quecktel module that integrates NB-IoT and GPS functionality. Since GPS and NB-IoT consume a lot of power, these two functionalities are disabled for most of the time and programmatically enabled only when needed (i.e., when an emergency call is made and the device is out of Bluetooth range of HomeTab). The Quecktel module is connected to a SIM card, which is required for NB-IoT functionality. These components are connected to a rechargeable Li-ion battery, which can be recharged using a wireless (induction) charger. The diagram of the safety watch circuit can be seen in Figure 1.

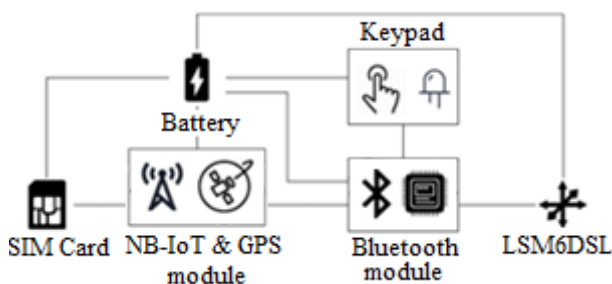


Figure 1: Diagram of safety watch circuit

The outer side of the safety watch housing is comprised of a membrane keypad used for manual alarm triggering (e.g. if the individual is in a dangerous situation). The keypad also integrates a small LED, used to provide a feedback to the users (e.g., alarm triggered, low battery alerts). Its appearance is shown in Figure 2.



Figure 2: Safety watch appearance

From a software perspective, the design principle behind the safety watch is to preserve the battery autonomy of the device. Therefore, the main processing unit is intended to sleep whenever possible and only wakes up when certain events happen, i.e., when there is an immediate danger for the user. The safety watch has two working modes, depending on whether the user wears the watch or not. If the watch is not worn, all working modes are disabled, since there is no need of motion monitoring, and only the device status (worn or not worn) is checked in 1-minute intervals. If the watch is worn, it monitors motion, accumulates the number of steps, and sends data over Bluetooth to the HomeTab. Once the battery of the device drops to 30% or lower, the sleeping time of the main processor increases from 5 to 10 minutes and the user is notified about the low battery level. The software design of the safety watch is illustrated in Figure 3.

The safety watch monitors users behaviour (activity levels), providing incentives to the users (through HomeTab) to move more and at the same time allow to determine unusually low activity (due to sickness). The integrated LSM6DSL step-count functionality enables the number of steps to be detected throughout the day and to be sent in regular 15-minutes intervals via Bluetooth to the HomeTab. This gives information about the user's activity levels, which the system later analyses to detect possible irregularities in the user's behaviour (which can be caused by an undetected disease). For example, if a user is feeling ill (has a flue), he will likely stay in bed significantly longer than when healthy, so the lack of movement can be detected, and caregivers notified.

2.1 Fall Detection

Automatic fall detection is one of the most important modules running on the safety watch. A machine learning method that can automatically detect falls and similar dangerous situations was developed and implemented in the final software of the safety watch.

For training of machine learning models, we used a publicly available dataset that contained acceleration data from a wrist-worn device from 17 subjects [12]. It comprised 11 daily-life activities, including 5 types of falling, namely: walking, standing, sitting, picking up an object, laying, jumping, falling backwards, falling sideward, falling forwards using knees, falling forwards using hands, and falling sitting in an empty

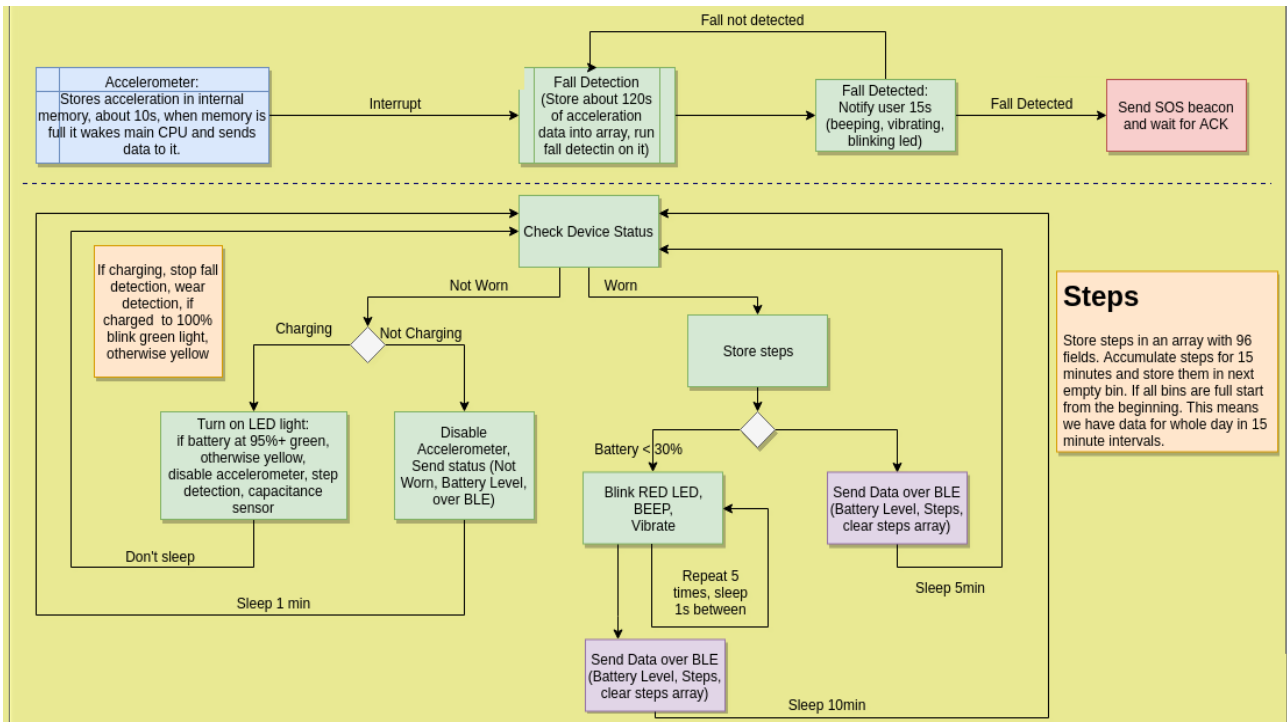


Figure 3: System software design

chair. Since our aim was to only detect falls in general, we grouped all fall-related activities as one class, and all other activities as another class. The non-fall activities were additionally under sampled, in order to adjust the class distribution of the dataset. The data were further segmented using a sliding window technique, with a window size of 2 seconds and 50% overlap between consecutive windows. To train the machine learning models, several statistical features were extracted from the acceleration signals, including mean, standard deviation, median, maximum, minimum, mean absolute change, variance, kurtosis, skewness, and similar. The window size and the optimal feature set was chosen based on our previous work [13].

Various machine learning algorithms were tested – Decision Tree (DT), Random Forest (RF), k-nearest neighbors (kNN). The different algorithms performances were evaluated using the leave-one-subject-out cross-validation technique. With this technique, the data is divided into N-number of folds (where N is the number of subjects in the dataset). Each fold is comprised of data from a single subject. In each iteration of the LOSO cross-validation, data from one subject is used for testing the method, and the training data is comprised of the remaining N-1 subjects. Among the tested algorithms, RF proved to have the best accuracies per watt of power consumed processing the data. RF is an ensemble classifier that fits a number of decision trees on various sub-samples of the dataset and outputs the majority class label from the constructed trees. It utilizes two random steps in the process of creating trees – a random sampling of the training data points and a random choosing of a splitting feature, which make it robust to noise and outliers [14]. The results achieved on the laboratory data with the best-performing RF

model, the kNN model and the DT model can be seen in Table 1, Table 2, and Table 3, respectively.

Table 1: Summed and normalized (per row) confusion matrix. LOSO evaluation with Random Forest model.

	Non-fall	Fall
Non-fall	97	3
Fall	2	98

Table 2: Summed and normalized (per row) confusion matrix. LOSO evaluation with Decision Tree model.

	Non-fall	Fall
Non-fall	91	9
Fall	8	92

Table 3: Summed and normalized (per row) confusion matrix. LOSO evaluation with kNN model.

	Non-fall	Fall
Non-fall	87	13
Fall	17	83

Since the aim of the system is to offer a great degree of accuracy in detecting actual fall, as well as in filtering false alarms, two metrics were analyzed: (i) sensitivity – capacity to detect actual fall, defined as the ratio between the number of falls correctly detected (true positives) and the falls that actually happened; (ii) specificity – capacity to filter false alarms,

defined as the ratio between properly discarded activities (true negatives) and the total number of discarded activities. From the confusion matrix presented in Table 1, it can be seen that the model has a very high sensitivity score – 98%, and specificity score – 97%. They are both very important for a real-life implementation of the model – it means that the model accurately detects falls, without triggering too many false alarms, which can be detrimental to users.

The implementation of the fall detection functionality on the hardware was also properly managed to extend the battery life. The most significant battery saving is done by processing the acceleration data in batches. The accelerometer stores acceleration values in its internal memory while the main processor sleeps. The accelerometer’s buffer fills in 10 seconds, and when it is full, it wakes the main processor, and the collected data is sent to it for further processing. The main processor stores for about 120 seconds of acceleration data before running the fall detection algorithm. Once the 120-seconds of data is stored, the required features are calculated from the acceleration signals, and the pre-trained RF model (stored in the RAM of the safety watch) is run. If no fall is detected in the two-minute segment, the main processor goes back to sleep, otherwise, an alarm procedure is triggered. The alarm is sent via Bluetooth to the HomeTab device, which forwards it to the server for further processing. If the safety watch is out-of-range of the HomeTab, it uses NB-IoT network for alarm transmission. In this case, it also tries to get the user’s location using a GPS signal.

3 Conclusion

This paper presented the design of a safety watch integrated into the HomeCare2020 comprehensive solution for a smart healthcare monitoring system, primarily targeted at elderly people. The main purpose of the safety watch is to help the elderly to live home independently until later age and to make them feel safer and more confident performing everyday tasks and activities. One of the most important modules running on the safety watch is fall detection, which makes the users able to call for emergency treatment in the case of a dangerous situation. For this purpose, different machine learning models were tested and compared. Among them, RF classification model proved to have the highest performance per watt of power consumed processing the data, which makes it the most suitable choice for implementation.

Overall, the software design of the system is highly energy-efficient and significantly extends the service time of the wearable device, which makes it convenient for use by elderly people. The system is easily operated and therefore shows great promise for providing long-term and continuous monitoring of the elderly in an unobtrusive way. We believe that it can efficiently contribute to improving remote healthcare services.

ACKNOWLEDGMENTS

The authors would like to thank everyone that helped in any way with producing this paper. The first author acknowledges the financial support from the Slovene Human Resources Development and Scholarship Fund – Ad Futura. Part of this research was done under EIT Health HomeCare2020 project.

REFERENCES

- [1] Roy, N.; Dubé, R.; Després, C.; Freitas, A.; Légaré, F. Choosing between staying at home or moving: A systematic review of factors influencing housing decisions among frail older adults. *PLoS One* 2018.
- [2] Institute of Medicine Falls in Older Persons: Risk Factors and Prevention. In *The Second Fifty Years: Promoting Health and Preventing Disability*; 1992 ISBN 978-0-309-04681-7.
- [3] Boyé, N. da; Van Lieshout, E. mm; Van Beeck, ed f.; Hartholt, K. a.; Van Der Cammen, T. jm; Patka, P. The impact of falls in the elderly. *Trauma* 2013.
- [4] Stevens, J.A.; Corso, P.S.; Finkelstein, E.A.; Miller, T.R. The costs of fatal and non-fatal falls among older adults. *Inj. Prev.* 2006.
- [5] Klaassen, B.; van Beijnum, B.J.F.; Hermens, H.J. Usability in telemedicine systems—A literature survey. *Int. J. Med. Inform.* 2016.
- [6] Majumder, S.; Aghayi, E.; Noferești, M.; Memarzadeh-Tehran, H.; Mondal, T.; Pang, Z.; Deen, M.J. Smart homes for elderly healthcare—Recent advances and research challenges. *Sensors (Switzerland)* 2017.
- [7] Sathyanarayana, S.; Satzoda, R.K.; Sathyanarayana, S.; Thambipillai, S. Vision-based patient monitoring: a comprehensive review of algorithms and technologies. *J. Ambient Intell. Humaniz. Comput.* 2018.
- [8] Benlamri, R.; Dockstader, L. MORF: A mobile health-monitoring platform. *IT Prof.* 2010.
- [9] Wu, W.; Cao, J.; Zheng, Y.; Zheng, Y.P. WAITER: A wearable personal healthcare and emergency aid system. In Proceedings of the 6th Annual IEEE International Conference on Pervasive Computing and Communications, PerCom 2008; 2008.
- [10] Peetoom, K.K.B.; Lexis, M.A.S.; Joore, M.; Dirksen, C.D.; De Witte, L.P. Literature review on monitoring technologies and their outcomes in independently living elderly people. *Disabil. Rehabil. Assist. Technol.* 2015.
- [11] nRF5 SDK - nordicsemi.com Available online: <https://www.nordicsemi.com/Software-and-tools/Software/nRF5-SDK> (accessed on Aug 26, 2020).
- [12] Martínez-Villaseñor, L.; Ponce, H.; Brieva, J.; Moya-Albor, E.; Núñez-Martínez, J.; Peñafort-Asturiano, C. Up-fall detection dataset: A multimodal approach. *Sensors (Switzerland)* 2019.
- [13] Gjoreski, H.; Stankoski, S.; Kiprijanovska, I.; Nikolovska, A.; Mladenovska, N.; Trajanoska, M.; Velichkovska, B.; Gjoreski, M.; Luštrek, M.; Gams, M. Wearable Sensors Data-Fusion and Machine-Learning Method for Fall Detection and Activity Recognition. In *Studies in Systems, Decision and Control*; 2020.
- [14] Breiman, L. Random Forest. *Mach. Learn.* 2001, 45, 5–32.

Machine Vision System for Quality Control in Manufacturing Lines

Ivana Kiprijanovska
Department of Intelligent
Systems
Jožef Stefan Institute,
Jožef Stefan International
Postgraduate School,
Ljubljana, Slovenia
ivana.kiprijanovska@ijs.si

Jani Bizjak
Department of Intelligent
Systems
Jožef Stefan Institute,
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
jani.bizjak@ijs.si

Samo Gazvoda
Cooking Appliances
Division
Gorenje Group
samo.gazvoda@gorenje.com

Matjaž Gams
Department of Intelligent
Systems
Jožef Stefan Institute,
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
matjaz.gams@ijs.si

ABSTRACT

In manufacturing, quality control is a process that oversees the aspects of production and ensures that only products that conform to industry standards and quality criteria leave the production line. Automation of the quality control process significantly reduces the time spent on products' testing, hence reducing the overall manufacturing costs. In this paper, we present a brief overview of the algorithms adopted to the aim of detection of one possible fault in the production of ovens – non-working oven fan. The detection is performed through visual data. In the initial experiments, several image processing algorithms were used, and the preliminary results are encouraging.

KEYWORDS

machine vision, image processing, fault detection

1 INTRODUCTION

Quality control is becoming an increasingly important aspect of today's manufacturing processes [1]. For efficient and successful production, manufacturers rely on quality control systems integrated into the manufacturing process. The traditional quality control process requires vast capacities of specialized labor. High utilization of the specialists may lead to human errors, low reliability of the process, and a negative impact on the quality of production. Compared to manual quality control, automated quality control systems offer a reliable control process with various other advantages, including the ability to work 24 hours a day and, in some tasks, perform faster measurements with higher accuracy and consistency compared to humans [2]. Such systems are also a practical choice when the test cases need to run regularly over a significant amount of time. Machine vision quality control systems play a growing role in modern manufacturing quality control systems. These systems rely on digital sensors inside

industrial cameras with specialized optics to acquire images [3]. After an image is acquired, computer hardware and software process, analyze, and measure various characteristics of the image for automated decision-making.

Development of an integrated system for comprehensive quality control in production with an intelligent process control system is the main aim of the ROBKONCEL project [4]. One of the objectives of this project is the detection of faults in the production of ovens. In this paper, we present the initial experiments in the detection of one of the possible faults – non-working oven fan.

2 PROBLEM DEFINITION

The quality control of the ovens is intended to take place in a factory environment, where products moving on a conveyor belt are visually observed, i.e., a machine vision system acquires videos of the ovens. These videos are segmented into image frames (at a 30 fps rate), and the obtained image frames are further processed to detect if the fan is working or not. For the initial experiments, we collected a few videos in a laboratory setting, with various lightings and camera positions, resulting in approximately 7200 images (~4000 working fan and ~3200 non-working fan). Additionally, the visual data of the ovens' fans were acquired through a closed door, which makes the fault detection more challenging (Figure 1). This is preferred as the process of opening and closing the door in a manufacturing environment would be too slow.



Figure 1: Image of an oven's fan acquired through a closed oven door.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
Information Society 2020, 5–9 October, 2020, Ljubljana, Slovenia
© 2020 Copyright held by the owner/author(s).

3 IMPLEMENTED TECHNIQUES

The image processing steps for the oven fault detection, i.e., non-working fan detection, are as following:

1. Object detection
2. Glare reduction
3. Image thresholding

Each of these steps and the image processing algorithms implemented in them are explained in the following sections.

3.1 Object Detection

In order to detect and isolate the circle area of the oven fan, we made use of the Hough Gradient Method [5], which is an extension to the standard Hough Transform technique [6] for isolating features of a particular shape within an image. The Hough Gradient Method is based on gradient information of edges and is used to improve the speed of the circle detection in order to meet real-time implementation requirements. The calculation steps of the Hough Gradient method are as follows: (i) detect edges in the image; (ii) calculate the local gradient for the edge points using a Sobel operator; (iii) use an accumulator to count the possible circle center on the normal direction of edge points' tangent; (iv) choose the peak circle center and circle radius for the general circle equation.

The implementation of the Hough Gradient method in OpenCV requires a single channel image, so the first step in the detection of circles was to convert the acquired images from the RGB color space to grayscale. Furthermore, two parameters of the circle detection function were tuned, namely: the minimum distance between the center coordinates of the detected circles and the ratio of the resolution of the original image to the accumulator resolution [5]. Before running the circle detection function, a simple median filter [7] was applied to the images for noise reduction. This helped in reducing the effects of various reflections in the glass part of the oven door. In general, without blurring, the algorithm tended to extract too many circular features, resulting in false circles detection. Therefore, this preprocessing step was crucial for successful circle detection. The circled detection algorithm resulted in a single circle detected in every image; however, with a varying radius. Since the further analyses require images with the same dimensions, the mean value of the detected circles' radius was calculated and used to isolate the fan area on the images. (Figure 2).

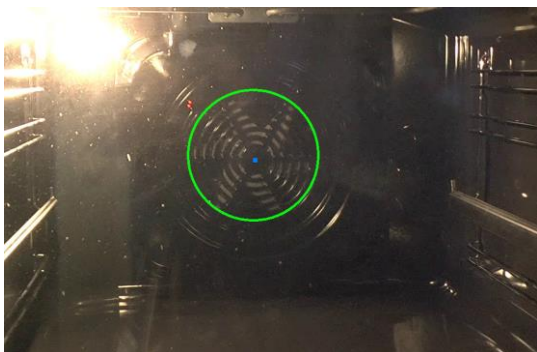


Figure 2: Detected oven's fan area.

3.2 Glare Reduction

A common problem in image processing is the occurrence of specular reflections on the images. In our case, since the videos of the fan were recorded through a glass, a significant amount of specular reflections, or glare, was produced during the recording. To reduce the effects of the glare, a glare reduction algorithm was applied. The basic glare reduction procedure consisted of 3 steps: (i) decomposition of the original image into a color, saturation and brightness component (HSV); (ii) finding particularly bright areas in the image; (iii) inpainting of these areas with the values of the surrounding pixels.

Each image was first converted into HSV color space, which describes the image by its hue (H), saturation (S) and brightness (V) component (Figure 3).

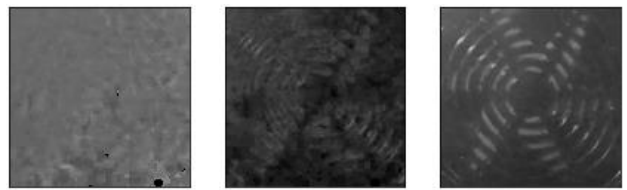


Figure 3: Image decomposition into hue, saturation and brightness component.

With such decomposition, a general rule for pixels that are subject to specular reflections can be derived; namely, an image can only contain glare if its color is not saturated, and it has high brightness. Since light reflections are white, any pixel containing glare cannot have saturation (since white has no color or saturation). Accordingly, we first filtered out the areas that have low saturation. Next, the area of the non-saturated pixel was reduced by an erosion operation, and the brightness values of the saturated pixels were set to 0. By filtering out the very bright pixels (e.g., all pixels that have a value larger than 130), we obtained the final glare mask (Figure 4).



Figure 4: Original image and the obtained glare mask.

The glared pixels were then interpolated with an inpainting operation. This operation fills the masked pixels with the values that stem from the adjacent non-masked pixels. The original image and its corrected version after the reduction of the glare can be seen in Figure 5. There is a significant amount of glare on the original image, which was effectively removed in the corrected image. The corrected image is a good approximation of the original image when no glare is present.

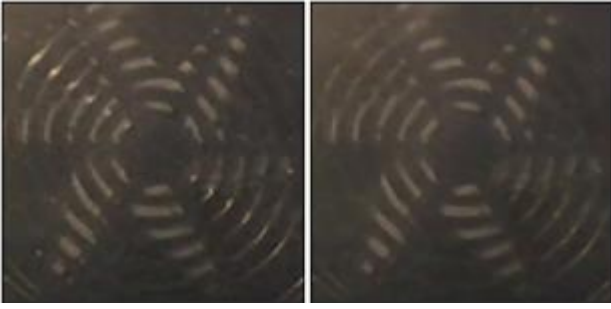


Figure 5: Original image and its corrected version.

3.3 Thresholding

If the two figures representing working and non-working fan in Figure 6 are analyzed, it can be seen that lighting allows the oven fan parts to stand out and be clearly seen behind the grid when the fan is not working. On the other hand, when the fan is working, the fan area behind the grid is blurred. Therefore, a simple thresholding method was utilized to distinguish working and non-working fan.



Figure 6: Working and non-working oven fan.

Thresholding is one of the simplest methods for image segmentation and creation of binary images [6]. The main goal of the utilized binary thresholding was to enhance the parts of the oven fan when it is not working. For that purpose, the images were firstly converted from RGB color space to grayscale. Next, with the binary thresholding method, each pixel in the images was replaced with a black pixel if its intensity was less than a chosen constant ($T=90$), or a white pixel if its intensity was greater than the chosen constant. This results in the illuminated parts of the oven fan becoming completely white (when the fan is not working), while the grid and the moving fan become completely black, as can be seen in the examples in Figure 7.

As a final step, the number of white pixels in the final binary-threshold images, which present only the non-working fans, was calculated. Then, the 5th percentile of these values was calculated and set as a threshold value when deciding if a given image represents a working or non-working fan. Basically, if the image contains more than X white pixels, where X is the previously calculated value of the 5th percentile, it is classified as a non-working oven fan; otherwise, it is classified as a working oven fan.

In the last post-processing step, the class for each image frame was taken as the majority class of the last 20 frames. It

helped in eliminating quick 1-frame changes from working to non-working, or vice versa.

Eventually, the implemented image-processing method resulted in 95% of correctly classified images, on four different videos. The confusion matrix of the method is presented in Table 1.

Table 1: Confusion matrix for the proposed method.

	Non-working	Working
Non-working	3117	82
Working	280	3720

As the main purpose of the system is to offer a high accuracy in detection of oven faults, while filtering false alarms, we additionally analysed two metrics: (i) sensitivity, i.e., method's capacity to detect actual faults (non-working fans), defined as the ratio between the number of non-working fan images correctly identified (true positives) and the total number of non-working fan images; (ii) specificity, i.e., method's capacity to filter false alarms, defined as the ratio between properly discarded images (true negatives) and the total number of discarded images. The method has a very high sensitivity score of 97%, and specificity score of 93%.



Figure 7: Non-working and working oven fan – thresholded images

4 CONCLUSION

In this paper, we presented an image processing pipeline adopted for the aim of detection of a possible fault in production of ovens – non-working oven fan. The image processing steps contain object detection (for isolating the oven fan area from the images), glare reduction (for reducing the effects of specular reflections), and image thresholding (for final decision-making). The preliminary results show that a quality control system that exploits image processing algorithms could be used in an automated manufacturing environment. In the future, we plan to employ reflection removal algorithms, which can significantly facilitate the object detection process, such as Sparse Blind Separation with Motions (SPBS-M) [8], Superimposed Image Decomposition (SID) [9], Ghosting Cues [10] and similar. However, the utilization of such algorithms may significantly impact the time performance of the method, so an acceptable trade-off between method's accuracy and time performance should be explored in future analyses.

ACKNOWLEDGMENTS

The first author acknowledges the financial support from the Slovene Human Resources Development and Scholarship Fund – Ad Futura. Part of this research was done under and for ROB KONCEL project.

REFERENCES

- [1] Mohamad, H.; Jenal, R.; Genas, D. Quality Control Implementation in Manufacturing Companies: Motivating Factors and Challenges. In Applications and Experiences of Quality Control; 2011.
- [2] Heleno, P.; Davies, R.; Brazio Correia, B.A.; Dinis, J. A machine vision quality control system for industrial acrylic fibre production. EURASIP J. Appl. Signal Processing 2002.
- [3] Golnabi, H.; Asadpour, A. Design and application of industrial machine vision systems. Robot. Comput. Integr. Manuf. 2007.
- [4] ROB KONCEL Available online: http://www.smm.si/?post_id=4682&lang=en (accessed on Aug 28, 2020).
- [5] Yuen, H.K.; Princen, J.; Dlingworth, J.; Kittler, J. A Comparative Study of Hough Transform Methods for Circle Finding.; 2013.
- [6] Shapiro, L.; Stockman, G. Computer Vision 1st Edition; 2001; ISBN 9780130307965.
- [7] Huang, T.S.; Yang, G.J.; Tang, G.Y. A Fast Two-Dimensional Median Filtering Algorithm. IEEE Trans. Acoust. 1979.
- [8] Gai, K.; Shi, Z.; Zhang, C. Blind separation of superimposed moving images using image statistics. IEEE Trans. Pattern Anal. Mach. Intell. 2012.
- [9] Guo, X.; Cao, X.; Ma, Y. Robust separation of reflection from multiple images. In Proceedings of the Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition; 2014.
- [10] Shih, Y.; Krishnan, D.; Durand, F.; Freeman, W.T. Reflection removal using ghosting cues. In Proceedings of the Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition; 2015.

Abnormal Gait Detection Using Wrist-Worn Inertial Sensors

Ivana Kiprijanovska
Department of Intelligent
Systems
Jožef Stefan Institute,
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
ivana.kiprijanovska@ijs.si

Hristijan Gjoreski
Faculty of Electrical Engineering
and Information Technologies
Skopje, N. Macedonia
hristijang@feit.ukim.edu.mk

Matjaž Gams
Department of Intelligent
Systems
Jožef Stefan Institute,
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
matjaz.gams@ijs.s

ABSTRACT

Falls are a major health problem among elderly people and often lead to serious physical and psychological consequences. Identification of elderly people who are at risk of falling helps for the selection of effective preventative measures that minimize the likelihood of falls. The occurrence of gait abnormalities is one of the most significant fall precursors. Wearable sensors enable continuous monitoring of gait during daily routines, and therefore offer the possibility of early detection of gait changes. In this paper, we analyze the ability of machine learning models to detect gait abnormalities using data from inertial sensors integrated into a smartwatch and how they perform on the dominant and non-dominant wrist.

KEYWORDS

Gait analysis, abnormal gait, fall risk assessment, smartwatch, wearable sensors

1 INTRODUCTION

Falls present a major health problem among elderly people. One-third of the population aged over 65 years experience at least one fall per year [1]. Falls greatly affect the quality of life and restrict the independence of those affected. They not only lead to severe physical consequences but also result in high health care costs. Due to the rapid aging of the population, this problem will further increase in the near future [2]. Therefore, there is an urgent need for reliable screening tools to identify those at risk and to target effective fall prevention strategies.

Falls are a consequence of several intrinsic and extrinsic fall risk factors, among which balance and gait disorders are the most common ones [3]. Gait is a sensitive indicator of an individual's overall health status, so the occurrence of abnormal gait patterns usually represents an early indication of an underlying neurodegenerative disorder. Clinical research has

shown that these disorders carry a high risk for falls, with an annual fall rate of 60–80% in patients with Alzheimer's, Parkinson's or similar diseases [4][5]. However, there is substantial evidence that falls can be prevented if individuals at increased risk of falling are identified and enrolled in targeted fall prevention programmes [6]. Therefore, identification of balance impairment and gait abnormalities is an essential step in fall prevention.

Camera-based 3D motion capture systems and instrumented walkways are considered as the gold standard in gait analysis in terms of accuracy. However, these systems are only suitable for hospitals or hospital-like settings, such as specialized gait analysis clinics, due to their size and the need for qualified professionals to operate them. Moreover, current clinical evaluation of gait is costly and time-consuming, and thus cannot be performed frequently. Even though the completeness and the accuracy of the clinical measurements are unquestionable, a mobile and pervasive gait analysis alternative suitable for non-hospital settings is a necessity. Recent technological advancements in wearable sensors offer means for analyzing gait during everyday-life living. Among wearable devices, wristbands and smartwatches are increasingly popular because people find the wrist placement one of the least intrusive placements to wear a device.

In this paper, we analyzed the ability of inertial sensors integrated into smartwatches to detect human gait abnormalities that are related to fall risk. Moreover, we studied how the performance of machine learning models on the non-dominant wrist compares to the performance on the dominant wrist.

2 RELATED WORK

The recent advancements in sensor technology have led to applications of wearable sensor devices in gait analysis for fall risk assessment. Several studies have been carried out by combining wearable devices with inertial sensors and machine learning methods. The general pipeline in these studies consists of signal acquisition while the person performs everyday-life activities or pre-defined functional tests, signal processing and feature engineering, and lastly training a machine learning classifier that produces an output that depends on the application.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
Information Society 2020, 5–9 October 2020, Ljubljana, Slovenia
© 2020 Copyright held by the owner/author(s).

Howcroft et al. [7] have presented insightful accounts of features, classification models and validation strategies related to sensor-based fall-risk assessment. They have found large heterogeneity in terms of sensor-based features and sensor placement. Regarding the features, the existing studies most often use features from the time and frequency domain, which include mean, variance and energy of the windowed inertial data, as well as spectral components such as dominant frequency and harmonic ratio [8]. As well as that, some biomechanical gait features, such as stride length, clearance, stance and swing time for each stride, cycle time, cadence and similar, have been revealed as effective predictors of falls [9].

In terms of the location of the sensors, the most exploited body positions are the shanks, waist, pelvis, and feet. In [10], the authors made use of wearable devices incorporating accelerometer and gyroscopes, worn on the shanks and waist. They proposed a general probabilistic modeling approach for classification of different pathological types of gait through the estimation of spatiotemporal features. They showed that a Support Vector Machine (SVM) classifier can identify mobility impairment in elderly people, with an accuracy of 90.5%. In [11], the authors showed that with assessment of walking quality during a six-minute walk test with accelerometers attached to their lower leg and pelvis, prospective fallers and non-fallers can be successfully differentiated with a Random Forest (RF) classifier. Similar findings were confirmed also for inertial sensors attached at the sternum, in [12]. However, these body locations may be found obtrusive for wearing a device for longer periods of time. On the other hand, the wrist is considered as the most unobtrusive and widely accepted position to wear a device, which does not affect everyday-life activities of the user. Still, sensors worn on the wrist are affected by frequent movements, as the hand is generally the most active part of the body. It makes the analysis of the gait very challenging, and thus wrist-worn devices have not yet been utilized for gait abnormalities detect for fall risk assessment. Considering the lack of evidence supporting the feasibility of fall risk assessment with sensors worn on the wrist, in this paper we analyze the performance of several machine learning methods that utilize inertial sensor data from a wrist-worn device.

3 DATASET

For this study, we collected a dataset comprised of recordings from 18 subjects (8 males, 10 females, aged 19-54). Each subject wore two smartwatches Mobvoi TicWatch E [13], one on the left, and one on the right wrist (Figure 1). The two smartwatches had an Android application that collected data from the inertial sensors integrated into the devices, namely: accelerometer, gyroscope, and magnetometer, at a sampling frequency of 100 Hz.

The subjects were walking back and forth along a 15-meters straight line and performed two scenarios – normal walk and simulated abnormal walk. In the normal gait scenario, subjects walked at a comfortable pace and performed a natural gait, while in the simulated abnormal walk scenario, subjects walked while wearing impairment glasses [8]. The glasses were used to simulate the effects of impairment, including reduction of

peripheral vision, visual distortion, balance deficit, and similar. These effects alter the gait and are highly correlated with an increased risk of falls [3]. Both scenarios (normal and abnormal walk) were repeated by each subject five times, resulting in ten walking sessions per subject. An example of two motion samples from the sensors in the smartwatch worn on the right wrist of one subject is shown in Figure 2.



Figure 1: Equipment for data collection

4 METHOD

The machine learning method that we developed for this study consists of several steps: preprocessing of the acquired sensor signals – filtering and data segmentation, feature engineering and extraction from signal segments, and training of a classification model. In the first step, the raw IMU signals were filtered with a band-pass filter with cut-off frequencies in the range of 0.5 to 3.5 Hz [14], which allowed for reducing the frequencies outside of the range of frequencies related to human walking activity [15]. After the filtering step, the sensor signals were segmented using a sliding window. Since window size and the sliding parameter have to be tuned correctly for the task at hand, the windowing parameters were determined empirically. Eventually, we chose a window size of 8 seconds, with 50% overlap between consecutive windows.

To train a classification model, we extracted several features from the time and frequency domain, for each sensor signal. The tsfresh python package [16] allows general-purpose time-series feature extraction, which we exploited in generating more than 100 features per sensor stream. These features included the minimum, maximum, mean, variance, the correlation between axes, their covariance, skewness, kurtosis, the number of times the signal is above/below its mean, the signal’s mean change, and its different autocorrelations, among others. Additional subset of frequency-domain features was also calculated using the signal’s power spectral density (PSD), which is based on the fast Fourier transform (FFT), and included PSD energy, entropy, and binned distribution, the largest magnitude from the PSD (of the dominant frequency in the signal), and first four statistical moments of the PSD (mean, standard deviation, skewness, and kurtosis) [17][18].

We compared several different ML models that have all previously been proven suitable for human activities analysis:

1) Decision tree (DT) [19] is an algorithm that learns a model in the form of a tree structure with decision nodes with two or more branches, each representing values for a tested feature, and leaf nodes which represent a decision on the target class. In other words, it predicts the target class by learning decision rules from the training features.

2) Random forest (RF) [20] is an ensemble of decision tree classifiers. It creates multiple decision trees, each trained on a

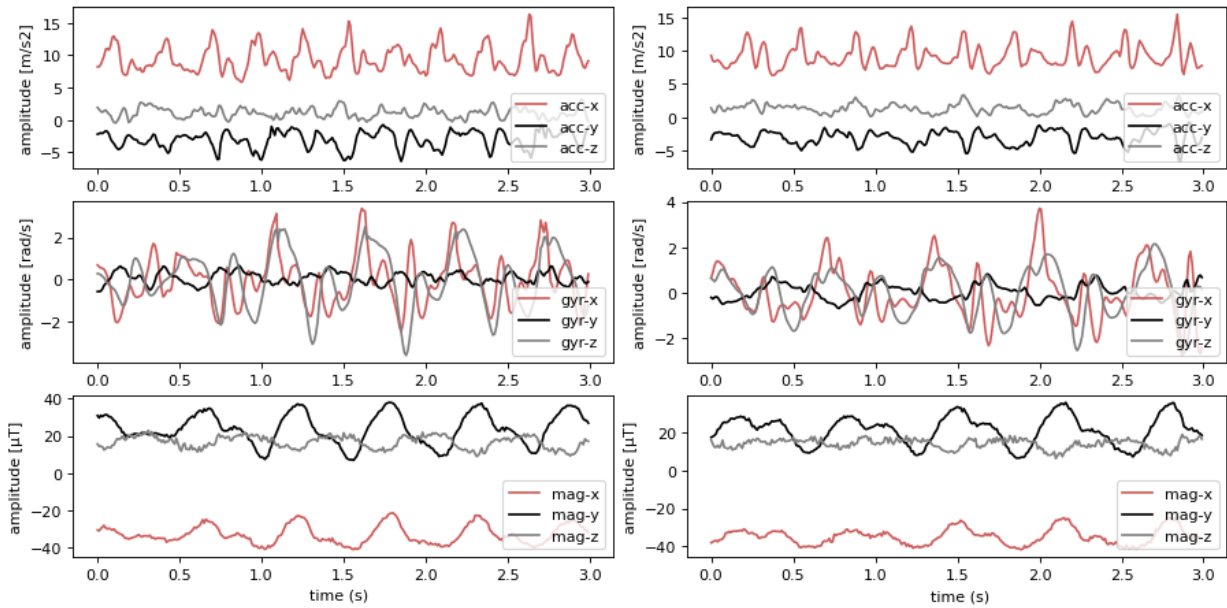


Figure 2: IMU sensors signals from normal walking session (left) and abnormal walking session (right)

bootstrapped sample of the original training data, and searches only across a randomly selected subset of features to determine a split. For the decision on the target class, each tree outputs a prediction, and the final prediction of the classifier is determined by a majority vote of the trees.

3) Support vector machine (SVM) [21] is an algorithm that is characterized by the use of kernel functions. They are used to transform feature vectors into higher dimensional space, in which a separation hyper-plane is learned to best fit the training data.

4) K-nearest neighbors (kNN) [22] is an algorithm that uses feature-vector similarity, i.e., for each feature vector in the test data, it finds the k-nearest neighbors in the training set. The final prediction of the classifier is determined by a majority vote of the chosen neighbors.

To estimate the generalization accuracy of the models, we utilized the leave-one-subject-out cross-validation technique. With this approach, the data is repeatedly split according to the number of subjects in the dataset. In each iteration, one subject is selected for testing purposes, while the other subjects are used for training the model. This procedure is repeated until data from all subjects have been used as test data.

5 EXPERIMENTAL RESULTS

To observe the performance of the models in real-life scenarios, we carried out several experiment. In fact, we observe the performance of the models on the left and right wrist separately, to see if they achieve similar result on both wrists.

Since real-life poses many challenges that should be taken into account, we considered three different training scenarios for each wrist. Namely, we test the accuracy of the models for six train-test combinations: training on the left wrist and testing on the right wrist (L - R), training on the right wrist and testing on the right wrist (R - R), training on both wrists and testing on the right wrist ((L+R) - R), training on the left wrist and testing

on the left wrist (L - L), training on the right wrist and testing on the left wrist (R - L), and training on both wrists and testing on the left wrist ((L+R) - L). With these combinations, we want to see if training a model with data from only a particular wrist or both wrists combined leads to higher accuracy. Moreover, another challenge that we took into account is a device with a model developed for the right (left) wrist to be worn on the left (right) wrist, hence the “switching wrists” combinations [23]. The results from these experiments can be seen in Table 1. The performance of the machine learning models is additionally compared with the performance of a baseline method - majority vote classifier.

From the presented results, it can be seen that the RF algorithm significantly outperforms the other algorithms for each train-test combination, while the kNN achieves the lowest accuracy in detection of gait abnormalities. Moreover, the results show that the right-left combination achieves 72.2% accuracy, which is significantly lower than the left-left combination, which achieves 83.9% with the RF model. On the other hand, the difference between the left-right and right-right combinations is minor – only 1.5 percentage point. These results suggest that models trained with data from the left wrist could perform well on both wrist, but the data acquired from the right wrist does not bring enough information to train a reliable model that could perform well on the left wrist, as well.

However, the problem of “switching wrists” could be overcome if the models are trained with data from both wrists. In fact, the models trained with data from the left and right wrist combined, outperform the other two combinations for both wrists, achieving the highest accuracy of 84.3% for the left wrist, and 82.3% for the right wrist with the RF model.

Overall, the results suggest that the models perform better for the left wrist. Since all subjects included in the dataset were right-handed, we can conclude that the non-dominant hand brings more information regarding the walking patterns of the subjects.

Table 1: Gait abnormality detection accuracy of individual classifiers.

Classifier	L - L	R - L	(L + R) - L	R - R	L-R	(L + R) - R
Baseline – Majority Classifier	61.4	61.4	61.4	61.4	61.4	61.4
DT	75.1	51.2	78.0	74.5	65.6	76.6
RF	83.9	72.2	84.3	82.8	81.3	84.3
SVM	68.3	61.0	72.4	64.4	66.4	71.4
kNN	63.2	57.3	63.8	61.2	62.6	63.0

6 CONCLUSION

In this paper, we analyzed the ability of machine learning algorithms to detect gait abnormalities using data from inertial sensors integrated into a smartwatch. Among the compared machine learning algorithms, Random Forest achieved the highest accuracy. The analysis of the performance of the models on the left and right wrist showed that they perform better on the left wrist, which was the non-dominant for the subjects included in the dataset. The experiments with the “switching wrist”, i.e., training the models with data collected from one wrist and testing on the other showed that the accuracy of the models significantly drops. However, when the models were trained with data from both wrists and applied on each wrist individually, the accuracy increased, outperforming even the models that were trained and tested on the same wrist. Therefore, the best practical solution is to deploy a model trained with data from both wrists. Overall, the results are satisfactory and show that data generated by wrist-worn inertial sensors is sufficient for gait abnormalities detection and can be used for fall risk assessment in non-clinical environments.

ACKNOWLEDGMENTS

The authors would like to thank all the participants that took part in the dataset collection. The first author acknowledges the financial support from the Slovene Human Resources Development and Scholarship Fund – Ad Futura.

REFERENCES

- [1] Dionyssiotis, Y. Analyzing the problem of falls among older people. *Int. J. Gen. Med.* 2012.
- [2] Ageing and health Available online: <https://www.who.int/news-room/fact-sheets/detail/ageing-and-health> (accessed on Aug 30, 2020).
- [3] Salzman, B. Gait and balance disorders in older adults. *Am. Fam. Physician* 2011.
- [4] Horikawa, E.; Matsui, T.; Arai, H.; Seki, T.; Iwasaki, K.; Sasaki, H. Risk of falls in Alzheimer’s disease: A prospective study. *Intern. Med.* 2005.
- [5] Allen, N.E.; Schwarzel, A.K.; Canning, C.G. Recurrent falls in parkinson’s disease: A systematic review. *Parkinsons. Dis.* 2013.
- [6] Institute of Medicine Falls in Older Persons: Risk Factors and Prevention. In *The Second Fifty Years: Promoting Health and Preventing Disability*; 1992 ISBN 978-0-309-04681-7.
- [7] Howcroft, J.; Kofman, J.; Lemaire, E.D. Review of fall risk assessment in geriatric populations using inertial sensors. *J. Neuroeng. Rehabil.* 2013.
- [8] Riva, F.; Toebes, M.J.P.; Pijnappels, M.; Stagni, R.; van Dieën, J.H. Estimating fall risk with inertial sensors using gait stability measures that do not require step detection. *Gait Posture* 2013.
- [9] Tunca, C.; Pehlivan, N.; Ak, N.; Arnrich, B.; Salur, G.; Ersoy, C. Inertial sensor-based robust gait analysis in non-hospital settings for neurological disorders. *Sensors (Switzerland)* 2017.
- [10] Mannini, A.; Trojaniello, D.; Cereatti, A.; Sabatini, A.M. A machine learning framework for gait classification using inertial sensors: Application to elderly, post-stroke and huntington’s disease patients. *Sensors (Switzerland)* 2016, 16.
- [11] Drover, D.; Howcroft, J.; Kofman, J.; Lemaire, E.D. Faller classification in older adults using wearable sensors based on turn and straight-walking accelerometer-based features. *Sensors (Switzerland)* 2017.
- [12] Brodie, M.A.; Lord, S.R.; Coppens, M.J.; Annegarn, J.; Delbaere, K. Eight-week remote monitoring using a freely worn device reveals unstable gait patterns in older fallers. *IEEE Trans. Biomed. Eng.* 2015.
- [13] TicWatch S&E - A smartwatch powered by Wear OS by Google Available online: <https://www.mobvoi.com/eu/pages/ticwatchse> (accessed on Aug 30, 2020).
- [14] Dehzangi, O.; Taherisadr, M.; ChagalVala, R. IMU-based gait recognition using convolutional neural networks and multi-sensor fusion. *Sensors (Switzerland)* 2017.
- [15] Antonsson, E.K.; Mann, R.W. The frequency content of gait. *J. Biomech.* 1985.
- [16] Overview on extracted features — tsfresh 0.16.1.dev65+gd190be5 documentation Available online: https://tsfresh.readthedocs.io/en/latest/text/list_of_features.html (accessed on Aug 30, 2020).
- [17] Su, X.; Tong, H.; Ji, P. Activity recognition with smartphone sensors. *Tsinghua Sci. Technol.* 2014.
- [18] Gjoreski, M.; Janko, V.; Slapničar, G.; Mlakar, M.; Reščič, N.; Bizjak, J.; Drobnic, V.; Marinko, M.; Mlakar, N.; Luštrek, M.; et al. Classical and deep learning methods for recognizing human activities and modes of transportation with smartphone sensors. *Inf. Fusion* 2020.
- [19] Gordon, A.D.; Breiman, L.; Friedman, J.H.; Olshen, R.A.; Stone, C.J. Classification and Regression Trees. *Biometrics* 1984.
- [20] Breiman, L. Random Forest. *Mach. Learn.* 2001, 45, 5–32.
- [21] Cortes, C.; Vapnik, V. Support-Vector Networks. *Mach. Learn.* 1995.
- [22] Aha, D.W.; Kibler, D.; Albert, M.K. Instance-Based Learning Algorithms. *Mach. Learn.* 1991.
- [23] Gjoreski, M.; Gjoreski, H.; Luštrek, M.; Gams, M. How accurately can your wrist device recognize daily activities and detect falls? *Sensors (Switzerland)* 2016.

Avtomatska detekcija obrabe posnemalnih igel

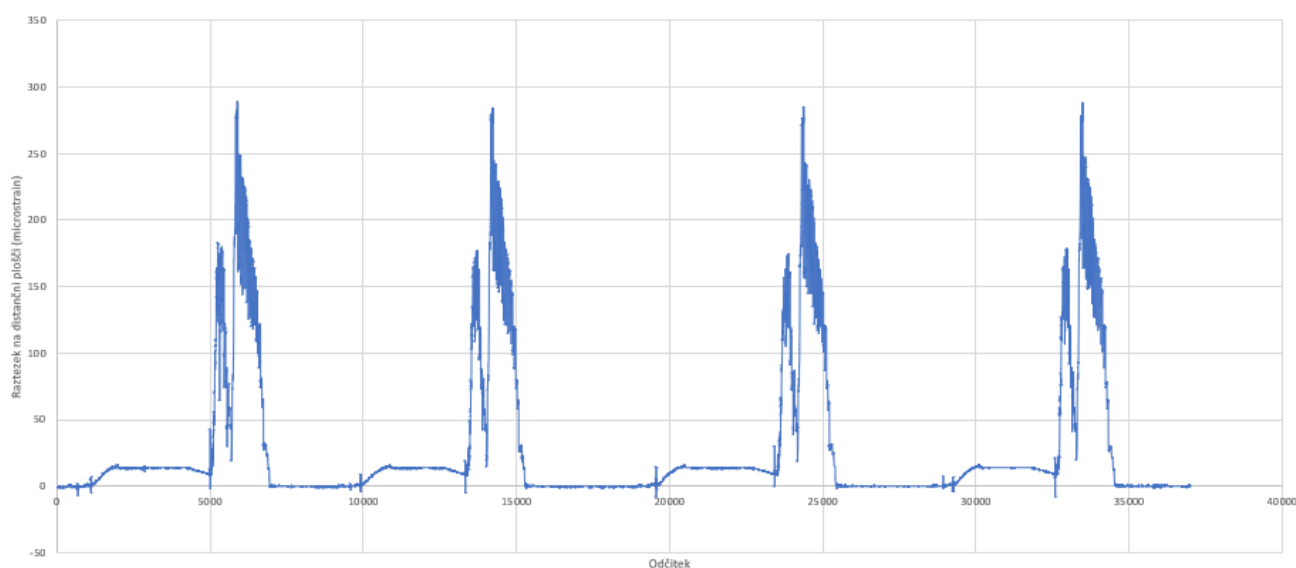
Automatic Wear Detection of Broaches

Primož Kocuvan
primoz.kocuvan@ijs.si
Institut "Jožef Stefan"
Jamova cesta 39
Ljubljana, Slovenija

Stefan Kalabakov
stefan.kalabakov@ijs.si
Institut "Jožef Stefan"
Jamova cesta 39
Ljubljana, Slovenija

Jani Bizjak
jani.bizjak@ijs.si
Institut "Jožef Stefan"
Jamova cesta 39
Ljubljana, Slovenija

Matjaž Gams
matjaz.gams@ijs.si
Institut "Jožef Stefan"
Jamova cesta 39
Ljubljana, Slovenija



Slika 1: Odčitki signala posnemalne igle

POVZETEK

Posnemanje materiala je ena izmed metod strojnega obdelovanja izdelkov, ki jih dosežemo s t.i posnemalno iglo. V grobem ločimo zunanje posnemanje in notranje posnemanje materiala. V prispevku se posvečamo notranjemu posnemanju, pri katerem se v začetku naredi manjšo luknjo v obdelovanec, nato pa postopoma oblikuje profil. To se doseže z različnimi premeri rezil tako, da je na začetku premer manjši, nato pa se postopno povečuje. Tako se lahko oblikuje poljuben krožni ali n-kotni profil. Zaradi obrabe rezil pri posnemanju se morajo le-ta redno menjati. V prispevku je opisan pristop napovedovanja obrabe posnemalne igle glede na cikel posnemanja. Glavna značilka, uporabljena za napovedovanje, je t.i mikroraztezanje (ang. microstrain), ki pove, za koliko se spremeni obremenitev na merilnem mestu v delcih na milijon. V prispevku je predstavljenih več metod strojnega učenja za reševanje omenjenega problema. Povprečna napaka

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

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

© 2020 Copyright held by the owner/author(s).

najboljše metode je 27 posnemanj oz. 1,8% glede na povprečno število posnemanj, ki se opravijo pred menjavo.

KLJUČNE BESEDE

Posnemalne igle, avtomatsko zaznavanje, regresija, strojno učenje

ABSTRACT

Broaching is one of the methods in metalworking, which is performed with the so-called broach. We distinguish between external broaching and internal broaching. In this paper, an internal broaching is presented, where a small hole is initially made in the workpiece, and then the broach gradually forms a profile. This is achieved with different blade diameters so that initially the diameter is smaller and then it gradually increases. Thus, any circular or polygon shape can be formed. Due to the wear during broaching, the blades must be replaced regularly. In this paper, an approach for predicting how many broaching processes or the number of work cycles can still be done before replacing the broach are presented. We did this by measuring and monitoring the microstrain parameter by the cut time, which tells how much the strain changes in parts per million. Thus, with regression

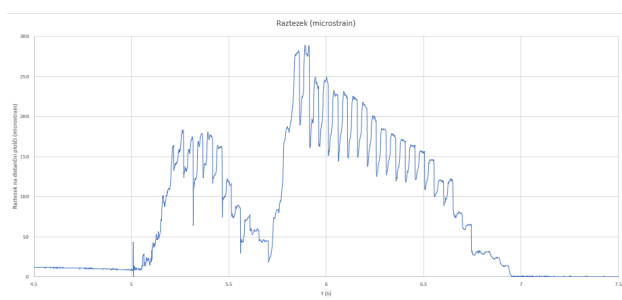
machine learning procedures, we learned a model that missed an average of 27 cycles or 1.8 %.

KEYWORDS

Broaches, automatic detection, regression, machine learning

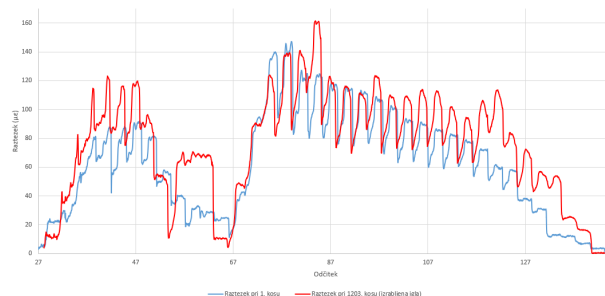
1 UVOD

Posnemanje materiala je zelo natančen postopek obdelovanja kovinskih izdelkov. Cena posnemalnih igel je relativno visoka (nekaj tisoč EUR), zato se posnemanje v industriji uporablja le v primerih, ko imamo dovolj veliko število obdelovancev. Uporaba obrabljene ali uničene posnemalne igle zaradi množične proizvodnje privede do visokih stroškov za proizvajalca, zato se igle trenutno menjajo po 1500 posnemanjih, ne glede na njihovo obrabo. S pomočjo strojnega učenja je mogoče natančneje napovedati, kdaj bo določena igla preveč obrabljena, s tem pa pridobimo boljši izkoristek igel ter takojšnje zaznavanje morebitne okvare igle. Avtorji prispevka so za razne industrijske aplikacije dobili več nagrad (prof. dr. Matjaž Gams [1]), medtem ko se je prvi avtor ukvarjal s procesiranjem časovnih signalov v svoji diplomski nalogi [2]. Nekateri raziskovalci so se lotili obdelave s pomočjo kombinacije strojnega vida in učenja [3], [4], [5] ter merjenja sil [6], [7]. Na sliki 2 je primer signala posnemanja enega cikla oziroma enega obdelovanca. Na abscisni osi je čas, medtem ko je na ordinatni osi obremenitev oziroma raztezek na distančni plošči (angl. microstrain) [8]. Distančna plošča je kovinska ploščica, ki zagotavlja ustrezen odmik med kovinskim izdelkom in posnemalno iglo. Tu merimo naš raztezek.

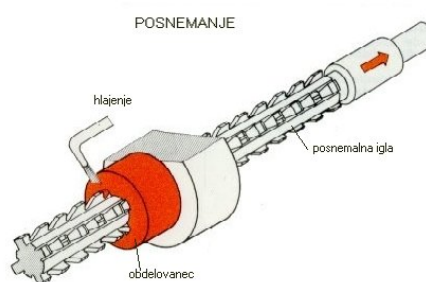


Slika 2: Primer povečave enega reza igle poljubnega signala

Opazimo, kako se raztezek (ki ga lahko interpretiramo kot silo) na distančni plošči spreminja, ko se spreminja premer zob posnemalne igle. Na sliki 3 je primer posnemalne igle (splošno), z rdečo barvo je označen obdelovanec. Smer puščice nakazuje pomik.



Slika 4: Primerjava raztezkov med obrabljeno in novo posnemalno iglo



Slika 3: Primer posnemalne igle [9]

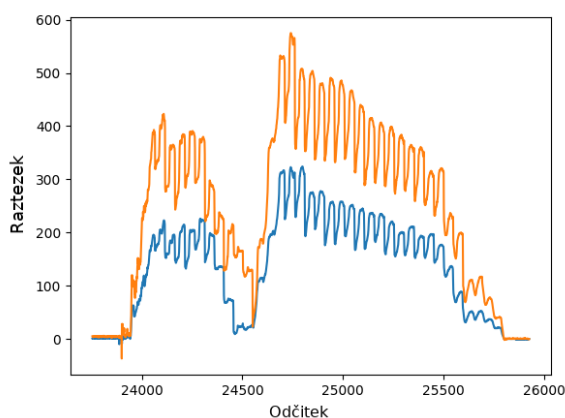
2 DEFINICIJA PROBLEMA

Rezila se med uporabo obrabljajo (postanejo topa), zaradi česar je potrebna večja sila za posamezen rez. S povečevanjem sile se večja verjetnost, da bo rez nepravilen, oz. se bo rezilo poškodovalo (npr. odlomil rezilni zob). Ko so rezila preveč obrabljena, jih je mogoče nabrusiti, kar je veliko ceneje od nakupa novega rezila v primeru nepopravljivih poškodb (npr. zloma zoba). Trenutno je postopek v proizvodnji tak, da se vsa rezila po 1500 posnemanjih zamenjajo, saj je verjetnost za napako po tem številu posnemanj previsoka. Problem je, da se rezilo zaradi različnih zunanjih dejavnikov (npr. mazivne tekočine, temperature itd.) obrablja hitreje ali počasneje, kar privede do okvar na izdelku ali slabšega izkoristka rezila.

Na sliki 4 je prikazana primerjava signala iz nove igle (modra) ter obrabljene igle (rdeča). Vidimo lahko, da ima posnemalna igla predstavljena z rdečim signalom v splošnem večji integral (površino pod krivuljo), to pomeni, da je sila večja. Razlikuje se tudi po številu ter jakosti posameznih vrhov, npr. v nekaterih primerih določeni vrhovi manjkajo (rezilo (nož) je popolnoma izrabljen).

3 REŠEVANJE PROBLEMA

Iz slike 4 lahko vidimo, da sta število in višina (integral) vrhov eden pomembnejših faktorjev pri prepoznavi okvare, sekundarni faktor pa je oblika vrhov. Avtomatskega prepoznavanja vrhov smo se lotili tako, da smo zaznali, kdaj se signal dvigne od standardne deviacije (šuma) signala. Med posameznimi rezi je igla v mirovanju, kar je razvidno iz slike 1. Na ta način smo dobili okno, ki vsebuje le signal, ki nastane med rezanjem. Poiskali smo okrog 1000 različnih atributov, ki opisujejo signal s pomočjo knjižnice Tsfresh [10]. Ti atributi so npr. minimalna in maksimalna vrednost signala, frekvence in vzorci, ki se pojavljajo v signalu. Nato smo attribute filtrirali z ozirom na relevantnost (prav tako



Slika 5: Primerjava odčitkov raztezka z leve in desne posnemalne igle

z omenjeno knjižnico), ki za vsak atribut izračuna p-vrednost oz. statistično stopnjo značilnosti. V zadnji fazi se nad množico p-vrednosti požene Benjamini-Yekutieli algoritem, ki se odloči katere značilke obdržimo in katere izločimo. Izkazalo se je, da so najpomembnejši atributi ploščina, maksimalna vrednost ter število vrhov, torej le trije atributi. Z izbranimi atributi smo s pomočjo strojnega učenja napovedali, v kakšnem ciklu oz. kako blizu okvari je določena igla. Uporabili smo naslednje pristope z učinkim okoljem Sci-kit learn [11], [12]:

- linearno regresijo (Linear Regression) [13],
- gradientno ojačitev za regresijo (Gradient Boosting) [14],
- klasifikator AdaBoost (AdaBoost Classifier) [15],
- K najbližjih sosedov (K Nearest Neighbours) [16].

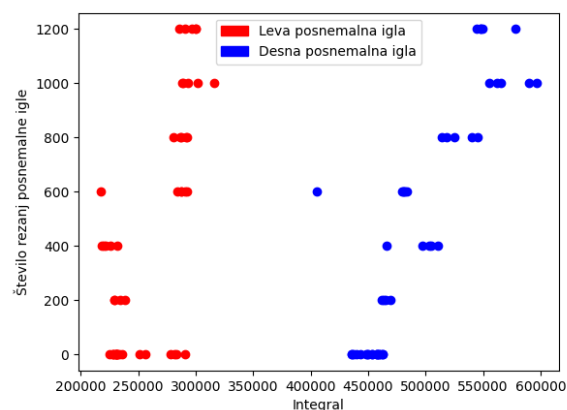
Obdelovalni oz. posnemalni stroj, s katerega smo pridobili meritve, ima levo in desno posnemalno iglo, pri čemer obe delujeta istočasno, torej obe posnemata (režeta) material hkrati. Na sliki 5 je primer meritev leve in desne posnemalne igle za posnemalnje ob določenem času. Opazimo, da ima ena igla večji integral, kar pomeni, da bi morali na začetku merilnega cikla kalibrirati iglo/senzor. S tem bi zagotovili enako izhodišče za nadaljnjo statistično obdelavo podatkov. Da bi se izognili tej težavi smo v tem prispevku primerjali le posnemalne igle, ki so na isti strani (leva ali desna).

4 REZULTATI

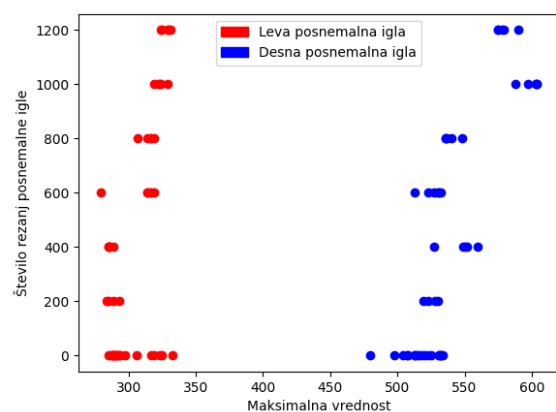
Na sliki 6 je prikazana primerjava integralov signala po določenem številu rezov (ordinata). Vidimo lahko (še posebej na desni igli), da se z večanjem števila rezov vrednosti integralov povečujejo, kar je skladno s pričakovanji, da je za enak rez s topim nožem potrebna večja sila.

Podobno, čeprav manj izrazito, lahko ugotovimo za maksimalno silo, ki nastane med rezom, kar je razvidno iz slike 7.

Na sliki 8 je prikazano število vrhov, ki jih algoritem prepozna. Po pričakovanjih je število vrhov obratno sorazmerno s številom rezov. Rezila na posameznih iglah se obrabljajo, zato te igle ne režejo več, torej je sila na rezilu nizka, saj igla ne postruži nič materiala. Nato sledi naslednja igla, ki ni obrabljena, ker predhodna igla ni opravila svojega dela, mora ta igla odstraniti večjo količino materiala, kar privede do večje sile ter obrabe na tem igli.



Slika 6: Integral signala glede na število rezanj posnemalne igle

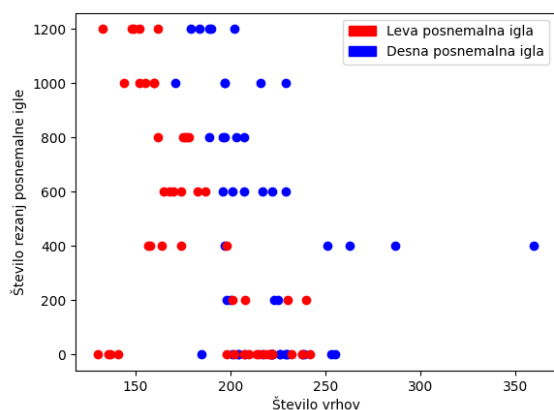


Slika 7: Maksimalne vrednosti signala glede na število rezanj posnemalne igle

Za napovedovanje zvezne vrednosti ciljne spremenljivke (regresija) uporabljamo metriko MAE (angl. Mean Absolute Error) in RMSE (angl. Root Mean Squared Error). Razlika je v tem, da metrika absolutne napake vrne le razliko absolutne napake, medtem ko RMSE vrne kvadrat te napake, s čimer kaznujemo večje razlike, torej primere, ko se napaka razlikuje za večje število ciklov. V našem primeru smo uporabili le vrednost MAE, ki je definirana z enačbo (1).

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - x_i| \quad (1)$$

V tabeli 1 so prikazani rezultati napovedovanja cikla za posamezno metodo strojnega učenja.



Slika 8: Število vrhov signala glede na število rezanj posnemalne igle

Tabela 1: Regresorji in njihove pripadajoče metrike MAE

Regresor	MAE
Linearna regresija	101,25
Gradient boost	27,58
AdaBoost	165,44
KNN	74,16

5 ZAKLJUČEK

Način merjenja mikroraztezka na distančni plošči ter analize časovnega signala s pomočjo strojnega učenja je naš prispevek na tem področju. Z navedenimi pristopi smo dobili povprečno absolutno napako (MAE) 27,58 kar pomeni, da se naš model v povprečju zmoti za 27,58, pri napovedovanju cikla trenutnega reza. Vrednosti (število ciklov) gre od 0 do 1500¹. To pomeni, da model s točnostjo 98,16 % napoveduje v katerem ciklu je posnemalna igla, oz. kdaj je iglo potrebno zamenjati. V nadaljevanju raziskave, se je potrebno osredotočiti na optimizacijo hiperparametrov posameznega regresorja. Končni cilj raziskave je implementacija tovrstnega primerjanja na podlagi signala v proizvodni proces.

ACKNOWLEDGMENTS

Ta raziskava je bila delno financirana s strani projekta ROB-KONCEL s šifro OP20.03530 in ARRS. Zahvaljujemo se podjetju UNIOR (Jože Ravničan in Tomaž Hohler).

LITERATURA

- [1] 2011. Ventil - revija za fluidno tehniko, avtomatizacijo in mehatroniko. V Ljubljana.
- [2] Primož Kocuvan. 2015. *Zaznavanje srčnega šuma v fonokardiogramih*. Diplomsko delo - Univerza v Ljubljani, 50.
- [3] Wenmeng Tian, Lee J. Wells in Jaime Camelio. 2016. Braaching tool degradation characterization based on functional descriptors. V (MSEC). 11th Manufacturing Science in Engineering Conference (MSEC2016), USA.

¹Model privzame, da je iglo potrebno zamenjati, ko signal izgleda, kot izgleda na igli s 1500 rezi. Če bi želeli točno izvedeti, kdaj je "točka preloma", torej ko je igla okvarjena, bi bilo potrebno izvesti še nekaj meritev/posnetkov, kjer bi se igla uporabljala dokler ne bi prišlo do napak na izdelku.

- [4] S.Kurada in C.Bradley. [n. d.] A machine vision system for tool wear assessment. 30, 295–304.
- [5] S.Damodarasamy in Shivakumar Raman. [n. d.] An inexpensive system for classifying tool wear states using pattern recognition. 170, 149–160.
- [6] Dongfeng Shi in Nabil N.Gindy. [n. d.] Tool wear predictive model based on least squares support vector machines. 21, 1799–1814.
- [7] S. Rangwala in D. Dornfeld. [n. d.] Sensor integration using neural networks for intelligent tool condition monitoring. 112, 219–228.
- [8] Anderson Langone Silva, Marcus Varanis, Arthur Guilherme Mereles, Clivaldo Oliveira in José Manoel Balthazar. [n. d.] A study of strain and deformation measurement using the arduino microcontroller and strain gauges devices. 41.
- [9] Srednja šola Koper. 2020. Posnemanje materiala. <http://www2.sts.si/arhiv/tehn/projekt3/Posnemanje/posnemanje.htm>.
- [10] Ts fresh library. 2020. Tsfresh. <https://tsfresh.readthedocs.io/en/latest/>.
- [11] Aurélien Géron. 2017. *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media; 1st Edition, 574.
- [12] Andreas C. Müller in Sarah Guido. 2016. *Introduction to Machine Learning with Python: A Guide for Data Scientists*. O'Reilly Media; 1st Edition, 400.
- [13] Sci kit learn. 2020. Regression - linear regression. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html.
- [14] Sci kit learn. 2020. Regression - gradient boost. <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html>.
- [15] Sci kit learn. 2020. Regression - adaboost. <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostRegressor.html>.
- [16] Sci kit learn. 2020. Regression k-nearest-neighbour. <https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html#sklearn.neighbors.KNeighborsRegressor>.

Povečevanje enakosti (oskrbe duševnega zdravja) s prepričljivo tehnologijo

Increasing Equality (in Mental Health Care) with Persuasive Technology

Tine Kolenik[†]

Odsek za inteligentne sisteme
Institut "Jožef Stefan" in
Mednarodna podiplomska šola
Jožefa Stefana
Ljubljana, Slovenija
tine.kolenik@ijs.si

Matjaž Gams

Odsek za inteligentne sisteme
Institut "Jožef Stefan"
Ljubljana, Slovenija
matjaz.gams@ijs.si

POVZETEK

Neuspešno spopadanje z naraščajočimi težavami z duševnim zdravjem močno ovira blaginjo posameznika in družbe. Kljub temu so ovire do dostopa in enakosti v oskrbi na področju duševnega zdravja, ki jih je veliko, znane, obsegajo pa od osebnih stigm do socialno-ekonomske neenakosti. Tehnologija, predvsem pa umetna inteligenca, ima ob takšnem stanju priložnost, da s svojim razvojem poskuša ublažiti obstoječi položaj z edinstvenimi rešitvami. Multi- in interdisciplinarne raziskave na področju prepričljive tehnologije, katere cilj je spreminjanje vedenja ali mentalnega stanja brez zavajanja in prisile, kažejo uspeh pri izboljšanju počutja pri ljudeh s tovrstnimi težavami. V prispevku so predstavljeni takšni sistemi s kratkim pregledom področja, glavni doprinos pa je analiza potencialnih težav in rešitev, ki jih prepričljiva tehnologija nudi na področju oskrbe duševnega zdravja. Zdi se, da prepričljiva tehnologija lahko dopolni obstoječe rešitve za pomoč pri duševnem zdravju, s tem pa zmanjša težave v dostopnosti in enakosti zdravstvene oskrbe kot tudi v enakosti nasploh.

KLJUČNE BESEDE

Digitalno duševno zdravje, prepričljiva tehnologija, umetna inteligenca, dostopnost in enakost zdravstvene oskrbe.

ABSTRACT

The inability to cope with increasing mental health issues among the populace severely hampers the well-being of both the individual and society. Barriers to access and equality in mental health care, many of which are well known, range from personal stigmas to socio-economic inequality. This offers technology, especially artificial intelligence, the opportunity to try to alleviate the existing situation with unique solutions. Multi- and interdisciplinary research in the field of persuasive technology, which aims to change behavior or mental states without deception and coercion, shows success in improving well-being

of people with mental health issues. This paper presents such systems with a brief overview of the field, with the main contribution being an analysis of potential problems and solutions that persuasive technology offers in the field of mental health care. Persuasive technology seems to be able to complement existing mental health care solutions, thereby reducing unequal access to and inequality in mental health care as well as reducing inequality in general.

KEYWORDS

Digital mental health, persuasive technology, artificial intelligence, mental health care access, equality.

1 UVOD

Težave na področju duševnega zdravja so že desetletja v porastu, uničujoč učinek tega pa so pripoznali tudi svetovni odločevalci, saj so Združeni narodi izboljšanje na tem področju uvrstili med svoje cilje trajnostnega razvoja [42]. Med temi težavami izstopajo predvsem stres, anksioznost in depresija (SAD). Beležijo, da se v nekaterih skupinah z akutnim stresom spopada 74% ljudi [24], z anksiozno motnjo 28% ljudi [5] in z depresijo 48% ljudi [36]. Kar se zdi še bolj problematično, je dejstvo, da v državah z nizkim in srednjim dohodkom okoli 80% ljudi ni deležno zdravljenja zaradi svojih duševnih težav, v državah z visokim dohodkom pa ta številka dosega okoli 35% [33]. Težave z duševnim zdravjem povzročijo daljnosežne in večplastne posledice, ki jih občutijo bolniki, njihova neposredna okolica (družina, skrbniki) in širša družba [41]. Bolniki se soočajo s slabšo kakovostjo življenja, nižjimi izobraževalnimi rezultati, nižjo produktivnostjo, potencialno revščino, socialnimi težavami in dodatnimi zdravstvenimi težavami. Skrbniki se soočajo z večjimi čustvenimi in fizičnimi izzivi, pa tudi z zmanjšanim dohodkom in povečanimi finančnimi stroški. Družba se vsako leto sooča z izgubo več odstotnih točk BDP in milijardami dolarjev na državo skupaj s poslabšanjem zaupanja v institucije javnega zdravja in s krhanjem socialne kohezije. Vse to vodi v čedalje močnejšo pozitivno povratno zanko – SAD ohranja in krepi SAD. Težave z duševnim zdravjem prepogosto vodijo tudi v izgubo človeškega življenja, saj se številne države spopadajo z visoko stopnjo samomorov [8]. Razlogi za višanje simptomov SAD vključujejo močno pomanjkanje strokovnjakov in predpisov za duševno zdravje [39] ter neenak dostop do oskrbe

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
Information Society 2020, 5–9 October 2020, Ljubljana, Slovenia
© 2020 Copyright held by the owner/author(s).

na področju duševnega zdravja [9]. Zato se zdi, da tehnološke in druge znanstvene terapevtske intervencije lahko pomagajo pri izboljšanju trenutnega stanja sistema, zlasti ker imajo posamezniki z duševnimi težavami terapije raje kot zdravila [2].

Zaradi napredka vedenjskih ved na področju človekovega odločanja in sorodnih pojavov [34] ter prihodom digitalnih tehnologij, umetne inteligence in velikega podatkovja se je razvoj usmeril v ustvarjanje tehnologij, ki bi pomagale, motivirale in usmerjale ljudi, da izboljšajo sebe in svet. Prepričljiva tehnologija (PT) je eden izmed rezultatov tovrstnih prizadevanj. Gre za tehnologijo, ki "spreminja stališča ali vedenja ali oboje (brez uporabe prisile ali zavajanja)" [12, str. 20]. Sprememba vedenja velja za pojav začasnega ali trajnega učinka na vedenje, odnos in druga duševna stanja posameznika v primerjavi s preteklostjo [12]. PT se že uporablja za pomoč pri duševnem zdravju [25, 27], kar prispeva k enakosti in omogoča lažji dostop do zdravstvene oskrbe [37].

Prispevek ima sledečo strukturo: poglavje 2 nudi pregled področja PT za pomoč pri duševnem zdravju, poglavje 3 analizira težave in rešitve, ki jih nudi PT, poglavje 4 pa poda nekaj zaključnih misli in idej za prihodnje delo.

2 PREGLED PODROČJA

Pričujoče poglavje vsebuje pregled področja PT in področja sprememb vedenja.

Sprememba vedenja je pojav, za katerega velja, da pri posamezniku povzroči začasen ali trajen učinek na njegovo vedenje v primerjavi s tem, kako se je vedel v preteklosti [12]. Ne vključuje le vedenja, temveč tudi duševna stanja. Intervencije za spremembo vedenja so velik del PT, ki se že pogosto uporablja na zdravstvenih področjih. Obstoječi sistemi s pomočjo umetne inteligence spremljajo vedenje ljudi ter njihova fiziološka in duševna stanja z namenom, da jih motivirajo in vplivajo na njihovo počutje, vse to pa lahko počnejo v naravnem jeziku [27].

Eden najpogosteje uporabljenih okvirjev prepričevanja in sprememb vedenja, ki jih uporabljajo takšne tehnologije, so Cialdinijeva načela prepričevanja (CPP) [6]. Obstajajo tudi drugi okviri [25, 27], vendar je za namene tega dela opisan samo CPP. Njegova glavna ideja je, da ne obstaja splošna strategija prepričevanja, ki bi delovala na vse ljudi. CPP zato opiše več strategij prepričevanja, saj so različni ljudje različno dovzetni za različne strategije.

CPP predvideva 7 strateških podlag za prepričevanje: 1) avtoriteta, ki cilja na ljudi, ki so bolj nagnjeni k temu, da jih motivira legitimna avtoriteta; 2) zavezanost in doslednost, ki sta namenjena ljudem, ki se bolj pogosto zavežejo k nečemu, če so se tako vedli že prej; 3) družbeni dokazi, ki ciljajo na ljudi, ki se ponavadi vedejo tako, kot se vedejo drugi; 4) všečnost, ki cilja na ljudi, za katere je bolj verjetno, da jih motivira nekdo, ki jim je všeč; 5) recipročnost, ki cilja na ljudi, ki so nagnjeni k vračanju uslug; 6) pomanjkanje, ki cilja na ljudi, ki menijo, da so redke stvari bolj dragocene; 7) enotnost, ki vpliva na ljudi, na katere vplivajo pozivi, ki se tičejo njihove skupinske identitete. Na različne ljudi vplivajo različne strategije, interaktivna tehnologija pa nudi orodje za učinkovitejšo izbiro tistih strategij, ki delujejo za določene ljudi.

Za izbiranje najučinkovitejše strategije se PT pogosto opira na osebnostne modele, kot je velikih pet faktorjev osebnosti [31], in vprašalnike za posamezne domene, kjer se PT uporablja (npr.

duševno zdravje). Osebnost se meri na različnih dimenzijah (odprtost, vestnost, ekstravertnost, sprejemljivost, nevroticizem), ki poskušajo opisati posameznikove tendence, povezane z njegovimi psihološkimi lastnostmi, kot so duševna stanja in odločanje. Prepričevanje na področju duševnega zdravja je hkrati bolj uspešno, če PT dostopa do podatkov o posameznikovem duševnem zdravju. V ta namen lahko uporabimo vprašalnike SAD [21] za kategorizacijo ljudi s simptomi SAD.

Okvirji prepričevanja so lahko implementirani v različne tehnološke platforme. Nedavni pregledni članek PT za zdravje in dobro počutje [27] je ugotovil, da so najpogosteje uporabljene platforme mobilne naprave (28%), sledijo igre (17%), spletna in socialna omrežja (14%) ter druge specializirane naprave (13%), namizne aplikacije (12%), senzorji in nosljive naprave (9%) ter zasloni v javnem prostoru (5%). Vrste aplikacij, ki delujejo kot PT, je na tem področju več, inteligentni kognitivni asistenti (IKA; znani tudi kot pogovorni roboti ali pogovorna umetna inteligenca) pa so najbolj napredni in razširjeni [4, 18, 26, 27, 30, 37, 44]. IKA izkazujejo številne človeku podobne sposobnosti, saj lahko do neke mere razumejo kontekst, se prilagajajo, se učijo, komunicirajo, sodelujejo, napovedujejo, zaznavajo, razlagajo in utemeljujejo. Najpomembneje je, da se IKA lahko pogovarjajo v naravnem jeziku in jih je zato mogoče ustvariti tako, da nudijo terapevtsko pomoč. Rezultati različnih preglednih člankov [4, 18, 26, 27, 30, 37] kažejo, da so IKA učinkovito sredstvo za lajšanje simptomov SAD. Izvedli smo kratek pregled prispevkov o najsodobnejših IKA za duševno zdravje in tri na kratko predstavljamo za ponazoritev tovrstne tehnologije. Vsi trije IKA [11, 14, 43] delujejo podobno, tako da z uporabo skriptiranih pogovorov in osnovnih sposobnosti procesiranja naravnega jezika nudijo pomoč. Ta je odvisna od uporabniškega modela, ki vsebuje podatke o čustvih uporabnikov in ravni SAD. Vsi IKA se v eksperimentih izkažejo za 15–20% uspešnejše pri lajšanju SAD od uradno priporočenega gradiva za samopomoč.

Takšna tehnologija nudi številne prednosti na področju duševnega zdravja: lahko je brezplačna in omogoča pomoč socialno-ekonomsko prikrajšanim ljudem; na voljo je 24 ur na dan, 7 dni v tednu, kar pomeni, da bolnikom ni treba čakati na naslednjo terapijo; veliko ljudi s simptomi SAD lažje zaupajo računalniku kot osebi [10, 22]; tehnologija je na voljo na oddaljenih lokacijah itd. Tehnologija lahko tako zmanjša obremenitev zdravstvenega sistema in njegovih izvajalcev ter zmanjša ovire za dostop do oskrbe duševnega zdravja na splošno. Pomembno je poudariti, da tehnologija deluje komplementarno in ne nadomešča strokovnjakov [16, 18, 37]. Prednosti rabe tovrstne tehnologije in morebitne težave so podrobneje obravnavane v naslednjem poglavju.

3 PREDNOSTI IN MOREBITNE TEŽAVE

Pričujoče poglavje obravnava posledice uporabe PT za duševno zdravje na področju spodbujanja enakosti in dostopnosti oskrbe duševnega zdravja, dotakne pa se tudi posledic na splošno. Posledice so razdeljene na tiste, ki ponujajo potencialne rešitve obstoječih težav in ovir za enakost in dostopnost, in tiste, ki se kažejo kot problemi te tehnologije pri doseganju enakosti. Na koncu poglavja so na kratko obravnavani tudi drugi problemi, ki na videz niso povezani z enakostjo, a so ključnega pomena, da PT doseže svoj potencial.

Kategorije, v katerih PT ponuja potencialne rešitve:

Stroški: Cena storitev, ki jih nudijo strokovnjaki za duševno zdravje (od psihoterapevtov do kliničnih psihologov in psihiatrov) se od države do države razlikujejo in so predvsem odvisni od državnih predpisov in subvencij. Neposredni stroški za bolnika so večinoma odvisni od števila strokovnjakov, ki so na voljo v določeni državi. Neodvisno od njihove višine pa stroški velikokrat ovirajo dostopnost do oskrbe ljudi iz nižjih socialno-ekonomskih okolij [23]. Dostop do PT za duševno zdravje je lahko brezplačen (in velikokrat je [11]) zaradi veliko nižjih stroškov, povezanih z izdelavo. K temu prispevajo trije glavni dejavniki: 1) razširljivost, kar pomeni, da lahko en sistem PT teoretično nudi pomoč neomejenemu številu ljudi (edini strošek, ki ga prinaša razširljivost, so stroški strožnika, ki so obrobni v primerjavi s človeškim delom) – nasprotno pa je en strokovnjak za duševno zdravje omejen na določeno število ljudi; 2) zmožnost, da učinkovit PT lahko ustvari veliko ljudi, predvsem zaradi obstoječih raziskav, ki temeljito poročajo o učinkovitih sistemih; in 3) količina ljudi, ki je sposobna proizvajati takšne sisteme, je veliko večja, kot je strokovnjakov, ki lahko ponudijo psihoterapevtsko in podobno pomoč.

Razpoložljivost: Problem razpoložljivosti lahko ločimo v tri podkategorije: 1) razpoložljivost na podlagi lokacije, 2) razpoložljivost na podlagi časa in 3) razpoložljivost na podlagi stroškov. Razpoložljivost na podlagi lokacije se nanaša na ljudi s težavami v duševnem zdravju na lokacijah, ki nimajo neposrednega dostopa do strokovnjakov za duševno zdravje (ali pa celo nimajo računalniškega dostopa do terapije na daljavo) [15]. Uporaba PT za duševno zdravje je ena redkih potencialnih rešitev v takih primerih. Razpoložljivost na podlagi časa se nanaša na ljudi z duševnimi težavami, ki potrebujejo terapevtsko pomoč v času, ko njihov izbrani strokovnjak ni na voljo. PT za duševno zdravje je na voljo 24 ur na dan, zato se njihova uporaba dopolnjuje z izbranim strokovnjakom za duševno zdravje. Bolniki nenehno poročajo o teh potrebah in take dopolnilne uporabe že obstajajo [29]. Razpoložljivost, ki temelji na stroških, se nanaša na ljudi z duševnimi težavami, ki potrebujejo terapevtsko pomoč, vendar nimajo sredstev za dostop, ki bi bil obsežnejši od najmanjše priporočene količine ur na teden [13] – ta se ocenjuje na eno uro na teden. Raziskave [13, 32] kažejo, da pogostejše terapije prinašajo boljše rezultate, dopolnilna uporaba PT za duševno zdravje pa lahko premosti to vrzel pri ljudeh, ki si ne morejo privoščiti več terapije. Razpoložljivost na podlagi stroškov je hkrati tesno povezana s širšim problemom stroškov, omenjenim v prejšnji kategoriji.

Stigma: Samostigma, predsodki, ki jih ljudje z duševnimi težavami imajo o sebi zaradi svojih težav, in javna stigma, odziv splošne populacije na ljudi z duševnimi boleznimi, predstavljata eno pglavitnih težav v boju proti duševnim težavam [7]. Težava je dvojna: zaradi javne stigme se posamezniki bojijo, kaj si bo družba mislila o njih, če bodo iskali zdravljenje, medtem ko se zaradi samostigme bojijo interakcije s strokovnjakom in dvomov, da si njihove težave pomoč sploh zaslužijo. Ta dvojnost prispeva k temu, da se posamezniki z duševnimi težavami odločijo, da se ne bodo zdravili pri strokovnjakih za duševno zdravje. Do 96% ljudi s SAD ne išče zdravljenja [35]. Raziskave o PT za duševno zdravje, zlasti o IKA za zdravljenje SAD, so pokazale, da ljudje v splošnem lažje zaupajo svoje težave računalniškemu ali mobilnemu sistemu kot osebi [22]. To je zato,

ker se ne bojijo, da bi jih obsojali, pridobijo pa zasebnost za razkrivanje svojih občutkov in misli na splošno. To pomeni, da se lahko število ljudi, ki se izogibajo stikom s strokovnjaki, zmanjša z uvedbo terapevtskih možnosti, za katere bolniki menijo, da so zanje varnejše in brez stigme.

Vendar pa takšna tehnologija potencialno prinaša tudi težave, ki jih je potrebno izpostaviti in resno obravnavati, da bi PT dosegel potencial, ki ga ima na področju duševnega zdravja:

Izključitev ranljivih skupin: Tehnološko usmerjene rešitve oskrbe duševnega zdravja lahko vodijo v izključevanje nekaterih ranljivih skupin. Mednje spadajo starostniki, najnižji socialno-ekonomski razred in kulturno specifične skupine. Zdi se, da je skupina, ki jo je uvedba tehnologije najbolj prizadela, skupina starostnikov [1]. Njihova nižja sposobnost vključevanja tehnologije v vsakdanje življenje lahko vodi v globlje razlike med njimi in drugimi generacijskimi skupinami. Druga skupina ljudi, ki je lahko izključena iz koristi PT za duševno zdravje, so ljudje iz najnižjega socialno-ekonomskega razreda, kjer jim PT morda sploh ne bo na voljo [28]. Poglobljanje že tako velikih razlik bi skupini povzročilo še bolj katastrofalne socialno-ekonomske življenjske razmere. Skupine, ki jih posvojitev tehnologije prizadene zaradi kulturnih razlik, so ključnega pomena pri razmisleku o napredku enakosti. Raziskave kažejo, da kulture z manj sodobnimi družbenopolitičnimi nagnjenji kažejo manjšo tendenco po posvajanju tehnologije [19]. Vseeno se zdi, da se večja prisotnost področja raziskovanja PT pojavlja tudi v nekaterih državah z nizkimi dohodki [40].

Pristranost v raziskovanju: Zaradi pomanjkanja standardov evalvacije PT za duševno zdravje je raziskovalno področje bolj dovzetno za pristranost v raziskovanju. Možnih težav je veliko: 1) sistemov PT, za katere se trdi, da so uspešni, ne preučujejo vedno v empiričnih poskusih (npr. randomizirana kontrolirana raziskava), temveč v kvazi eksperimentih [43] ali sploh ne; 2) metrika, na podlagi katere bi lahko ocenili takšne sisteme, ni jasna (običajno izhaja posredno iz njihove učinkovitosti v raziskavi, kjer je cilj lajšanje simptomov SAD [37]); 3) ni soglasja o tem, kateri podatki so potrebni, da sistem razume uporabnika in mu s tem nudi učinkovito pomoč, s čimer je izbira vrste podatkov zaenkrat večkrat odvisna od predpostavk raziskovalcev kot pa od obstoječih spoznanj.

Uporaba PT za duševno zdravje ima tudi težave, ki se ne nanašajo samo na doseganje enakosti in dostopnosti. Čeprav so izjemno pomembni, je njihova poglobljena analiza izven okvirjev tega dela. Vseeno jih nekaj omenimo: 1.) problem varstva osebnih podatkov [3]; 2) problem pomanjkanja longitudinalnih raziskav o spremembah vedenja s PT [20]; 3) etičnost uporabe osebnih podatkov za prepričevanje [17]; in 4) potencialni problem avtomatizacije in izgube zaposlitve strokovnjakov za duševno zdravje. Zagotovo obstajajo tudi druge težave in pomisleki, vendar smo želeli, da je ta seznam kratek in da z njim pokažemo, da obstajajo tudi druge težave s PT in da se jih zavedamo.

4 ZAKLJUČEK IN PRIHODNJE DELO

Pričujoče delo raziskuje, kako lahko prepričljiva tehnologija, ki poskuša brez prisile vplivati na vedenje ljudi, poveča enakost in dostopnost oskrbe duševnega zdravja, s čimer bi okrepila enakost

na splošno. Delo, ki se nadalje osredotoča na stres, anksioznost in depresijo, preučuje, zakaj je duševno zdravje precejšnja ovira za enakost in zakaj imajo ljudje z duševnimi težavami ovire pri dostopu do zdravstvene oskrbe. Nato poda svoje argumente za uporabo prepričljive tehnologije v tej domeni. Sledi predstavitev prepričljive tehnologije v njeni multi- in interdisciplinarni sestavi vedenjskih znanosti in računalništva ter umetne inteligence. Predstavljeni so primeri implementacije prepričljive tehnologije za duševno zdravje v inteligentnih kognitivnih asistentih, vključno z njihovo učinkovitostjo za lajšanje simptomov stresa, tesnobe in depresije. Delo nazadnje raziskuje potencialne rešitve, ki jih taka tehnologija ponuja na področju duševnega zdravja, in morebitne težave, ki bi jih lahko ustvarila. Prihodnje delo vključuje nadaljnje raziskovanje problemov in rešitev, poglobitev v tehnično zasnovo tovrstnih tehnologij, še posebej tistih, ki uporabljajo umetno inteligenco, ter ponujanje novih konceptualnih in tehničnih smernic za PT za duševno zdravje pri zmanjševanju neenakosti oskrbe duševnega zdravja in neenakosti na splošno.

ZAHVALA

Delo je nastalo v okviru programa mladih raziskovalcev, ki ga je financirala Javna agencija za raziskovalno dejavnost Republike Slovenije iz državnega proračuna.

VIRI

- [1] I. Amaral in F. Daniel, 2016. Ageism and IT: social representations, exclusion and citizenship in the digital age. *Lecture Notes in Computer Science* 9755 (2016), 159–166.
- [2] M. C. Angermeyer in H. Matschinger, 1996. The effect of personal experience with mental illness on the attitude towards individuals suffering from mental disorders. *Social Psychiatry and Psychiatric Epidemiology. The International Journal for Research in Social and Genetic Epidemiology and Mental Health Services* 31, 6 (1996), 321–326.
- [3] S. Avancha, A. Baxi in D. Kotz, 2012. Privacy in mobile technology for personal healthcare. *ACM Computing Surveys* 45, 1 (2012).
- [4] D. Bakker, N. Kazantzis, D. Rickwood in N. Rickard, 2016. Mental Health Smartphone Apps: Review and Evidence-Based Recommendations for Future Developments. *JMIR Mental Health* 3, 1 (2016).
- [5] A. Baxter, J.M. Scott, T. Vos in H. Whiteford, 2013. Global prevalence of anxiety disorders: a systematic review and meta-regression. *Psychological Medicine*, 43 (2013), 897–910.
- [6] R. Cialdini. 2016. *Pre-Suasion: A Revolutionary Way to Influence and Persuade*, Simonand Schuster. Simon & Schuster, New York, NY.
- [7] P. W. Corrigan in A. C. Watson, 2002. Understanding the impact of stigma on people with mental illness. *World psychiatry: official journal of the World Psychiatric Association (WPA)* 1, 1 (2002), 16–20.
- [8] S. C. Curtin, M. Warner in H. Hedegaard, 2016. *Increase in suicide in the United States, 1999-2014*. U.S. Department of Health and Human Services, Centers for Disease Control and Prevention, National Center for Health Statistics, Hyattsville, MD.
- [9] European Commission. 2018. *Inequalities in access to healthcare - A study of national policies*. <https://ec.europa.eu/social/main.jsp?catId=738&langId=en&pubId=8152>
- [10] A. Fadhil in G. Schiavo, 2019. Designing for Health Chatbots. *arXiv*, (2019). <https://arxiv.org/abs/1902.09022>
- [11] K. K. Fitzpatrick, A. Darcy in M. Vierhile, 2017. Delivering Cognitive Behavior Therapy to Young Adults With Symptoms of Depression and Anxiety Using a Fully Automated Conversational Agent (Woebot): A Randomized Controlled Trial. *JMIR Mental Health* 4, 2 (2017).
- [12] B. J. Fogg. 2002. *Persuasive technology*. MK, Burlington, MA.
- [13] N. Freedman idr., 1999. The Effectiveness of Psychoanalytic Psychotherapy: the Role of Treatment Duration, Frequency of Sessions, and the Therapeutic Relationship. *Journal of the American Psychoanalytic Association* 47, 3 (1999), 741–772.
- [14] R. Fulmer idr., 2018. Using Psychological Artificial Intelligence (Tess) to Relieve Symptoms of Depression and Anxiety: Randomized Controlled Trial. *JMIR Mental Health* 5, 4 (2018).
- [15] K. Gibson idr., 2009. Clinicians' attitudes toward the use of information and communication technologies for mental health services in remote and rural areas. *Canadian Society of Telehealth Conference*, Vancouver, October 3–6, (2009).
- [16] C.M. Kennedy, J. Powell, T.H. Payne, J. Ainsworth, A. Boyd in I. Buchan, 2012. Active Assistance Technology for Health-Related Behavior Change: An Interdisciplinary Review. *Journal of Medical Internet Research* 14, 3 (2012).
- [17] D. B. Klein, 2004. Statist Quo Bias. *Econ. Jour. Watch* 1 (2004), 260–71.
- [18] L. Laranjo idr., 2018. Conversational agents in healthcare: a systematic review. *Journal of the American Medical Informatics Association* 25, 9 (2018), 1248–1258.
- [19] S. G. Lee, S. Trimi in C. Kim, 2013. The impact of cultural differences on technology adoption. *Journal of World Business* 48, 1 (2013), 20–29.
- [20] S. S. Lee, Y. K. Lim in K. P. Lee, 2011. A long-term study of user experience towards interaction designs that support behavior change. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems*, ACM, New York, NW, 2065–2070.
- [21] S. H. Lovibond in Peter F. Lovibond. 1996. *Manual for the depression anxiety stress scales*. Psychology Foundation of Australia, Sydney.
- [22] G. M. Lucas, J. Gratch, A. King in L. P. Morency, 2014. It's only a computer: Virtual humans increase willingness to disclose. *Computers in Human Behavior* 37 (2014), 94–100.
- [23] P. McCrone idr., 2004. Cost-effectiveness of computerised cognitive-behavioural therapy for anxiety and depression in primary care: Randomised controlled trial *British Journal of Psychiatry* 185, 1 (2004), 55–62.
- [24] Mental Health Foundation. 2018. *Stress: Are we coping?* Mental Health Foundation, London.
- [25] D. C. Mohr idr., 2013. Behavioral intervention technologies: evidence review and recommendations for future research in mental health. *General hospital psychiatry* 35, 4 (2013).
- [26] J. L. Z. Montenegro, C. A. da Costa in R. da Rosa Righi, 2019. Survey of conversational agents in health. *Expert Systems with Applications* 129 (2019), 56–67.
- [27] R. Orji in K. Moffatt, 2016. Persuasive technology for health and wellness: State-of-the-art and emerging trends. *Health Informatics Journal* 24, 1 (2016), 66–91.
- [28] M. Pigato. 2001. *Information and communication technology, poverty, and development in sub-Saharan Africa and South Asia (English), Africa Region working paper series; no. 20*. The World Bank, Washington, D.C.
- [29] M. Price idr., 2013. mHealth: A Mechanism to Deliver More Accessible, More Effective Mental Health Care. *Clinical Psychology & Psychotherapy* 21 (2013), 427–436.
- [30] S. Provoost, H. M. Lau, J. Ruwaard in H. Riper, 2017. Embodied Conversational Agents in Clinical Psychology: A Scoping Review. *Journal of Medical Internet Research* 19, 5 (2017).
- [31] B. Rammstedt in O.P. John, 2007. Measuring personality in one minute or less: A 10-item short version of the Big Five Inventory in English and German. *Journal of Research in Personality* 41, 1 (2007), 203–212.
- [32] R. Sandell idr., 2000. Varieties of long-term outcome among patients in psychoanalysis and long-term psychotherapy: a review of findings in the Stockholm Outcome of Psychoanalysis and Psychotherapy Project (STOPP). *The International Journal of Psychoanalysis* 81 (2000), 921–942.
- [33] A. Schmidtke idr., 1996. Attempted suicide in Europe: rates, trends and sociodemographic characteristics of suicide attempters during the period 1989–1992. *Acta Psychiatrica Scandinavica* 93 (1996), 327–38.
- [34] R. H. Thaler in C. R. Sunstein. 2008. *Nudge: improving decisions using the architecture of choice*. Yale University Press, New Haven, CT.
- [35] G. Thornicroft idr., 2017. Undertreatment of people with major depressive disorder in 21 countries. *British Journal of Psychiatry* 210, 2 (2017), 119–124.
- [36] J. M. Twenge, 2014. Time Period and Birth Cohort Differences in Depressive Symptoms in the U.S., 1982–2013. *Social Indicators Research* 121, 2 (2014), 437–454.
- [37] A. N. Vaidyam idr., 2019. Chatbots and Conversational Agents in Mental Health: A Review of the Psychiatric Landscape. *Canadian journal of psychiatry* 64, 7 (2019).
- [38] P. S. Wang idr., 2007. Use of mental health services for anxiety, mood, and substance disorders in 17 countries in the WHO world mental health surveys. *The Lancet* 370, 9590 (2007), 841–50.
- [39] P. Winkler idr., 2017. A blind spot on the global mental health map: a scoping review of 25 years development of mental health care for people with severe mental illnesses in central and eastern Europe. *The Lancet Psychiatry* 4, 8 (2017), 634–642.
- [40] H. Winschiers-Theophilus idr., 2018. *Proceedings of the Second African Conference for Human Computer Interaction: Thriving Communities*. Association for Computing Machinery, New York, NY.
- [41] World Health Organization. 2003. *Investing in Mental Health*. <https://apps.who.int/iris/handle/10665/42823>
- [42] World Health Organization (WHO). 2013. *Mental Health Action Plan 2013-2020*. Geneva, Switzerland.
- [43] A. Yorita idr., 2018. A Robot Assisted Stress Management Framework: Using Conversation to Measure Occupational Stress. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*.
- [44] M. Mlakar, A. Tavčar, G. Grasselli in M. Gams. 2018. *Asistent za stres*. <http://poluks.ijs.si:12345/>.

Analiza glasu kot diagnostična metoda za odkrivanje Parkinsonove bolezni

Speech Analysis as a Diagnostic Method for the Detection of Parkinson's Disease

Andraž Levstek
Gimnazija Jožeta Plečnika
Šubičeva ulica 1
Ljubljana, Slovenija
levstek.andraz@gmail.com

Darja Silan
Gimnazija Jožeta Plečnika
Šubičeva ulica 1
Ljubljana, Slovenija
darja.silan@gjp.si

Aljoša Vodopija
Institut "Jožef Stefan"
Jamova cesta 39
Ljubljana, Slovenija
aljosa.vodopija@ijs.si

POVZETEK

Parkinsonova bolezen je neurodegenerativna bolezen, ki povzroča težave v delovanju mišic zaradi pomanjkanja dopamina v možganskem deblu, poleg tega vpliva tudi na glas. Slednji postane bolj monoton, hripav in šibek. Zaradi naštetih sprememb se za diagnosticiranje Parkinsonove bolezni vse pogosteje uporablja analiza glasu z metodami umetne inteligence. V tej raziskavi smo s pomočjo metod strojnega učenja primerjali zvočne posnetke glasu zdravih oseb in bolnikov s Parkinsonovo boleznijo. Za izboljšavo klasifikacijske točnosti smo dodatno uporabili pristop zmanjševanja razsežnosti. Najbolj točen klasifikator smo zgradili z uporabo metode naključnih gozdov, s katerim smo dosegli 73 % točnost. Dobljeni rezultati nakazuje na povezavo med Parkinsonovo boleznijo in karakteristično spremembo glasu. Ocenili smo pomembnost posameznih zvočnih posnetkov in pripadajočih atributov. Izsledke raziskave lahko uporabimo za nadgradnjo obstoječe metodologije s predlogi za dodatne posnetke, ki vsebujejo več informacij o prisotnosti Parkinsonove bolezni.

KLJUČNE BESEDE

Parkinsonova bolezen, analiza glasu, strojno učenje, naključni gozdovi, pomembnost atributov

ABSTRACT

Parkinson's disease is a neurodegenerative disorder that causes impaired muscle function because of a lack of dopamine in the brain stem. Parkinson's disease also affects speech ability. The voice becomes monotone, hoarse and feeble. For this reason, one of the emerging ways to diagnose Parkinson's disease is speech analysis using artificial intelligence. In this paper, we use machine learning to connect voice samples to the presence of Parkinson's disease. To improve the classification accuracy, we additionally use a dimensionality reduction approach. The most accurate classifier was built with random forest, with an accuracy of 73 %. The experimental results indicate the correlation between the voice changes and the presence of Parkinson's disease. Additionally, we estimate the importance of individual voice samples and corresponding features. The results can be used to improve the current methodology by proposing additional voice samples, that contain more information on the presence of Parkinson's disease.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

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

© 2020 Copyright held by the owner/author(s).

KEYWORDS

Parkinson's disease, speech analysis, machine learning, random forest, feature importance

1 UVOD

Parkinsonova bolezen je neurodegenerativno in izčrpavajoče bolezensko stanje, ki vpliva na osrednje živčevje. Bolezen prizadene približno 1 % ljudi, starejših od 60 let. Bolnik s Parkinsonovo boleznijo se pogosto trese, ima težave s hojo in ravnotežjem, njegovo gibanje postane počasno, pojavi se rigidnost. Pojavijo se lahko tudi duševne motnje, kot so anksioznost, depresija ter težave s spanjem, razmišljanjem in obnašanjem.

Parkinsonova bolezen vpliva tudi na glas. Večina bolnikov ima govorne težave, kot so šibek, zadihan, hripav, višji in monoton glas. Za bolnika so značilne hripavost, zmanjšana jakost glasu, težava s pravilno artikulacijo fonemov in brbljanje [5].

Diagnostične metode, ki bi stoodstotno dokazala prisotnost Parkinsonove bolezni, še ne poznamo. Diagnoza temelji na vidnih in razpoznavnih simptomih, preteklem zdravstvenem stanju, fizičnem ter nevrološkem pregledu in bolnikovi anamnezi [13]. Po kriterijih mora biti za dokaz Parkinsonove bolezni prisotna akineza ter še vsaj ena druga lastnost (npr. tremor rok pri mirovanju, rigidnost ali posturalne motnje). Po teh kriterijih se Parkinsonova bolezen lahko identificira z 90 % točnostjo, vendar diagnoza lahko traja več let [12]. Pri diagnosticiranju se uporablja tudi slikanje možganov z magnetno resonanco, pozitronsko emisijsko tomografijo in računalniško tomografijo. Vse našete diagnostične metode so drage ter zahtevne, zato se išče cenejše in preprostejše metode [13].

V diagnostične namene se vse pogosteje uporablja analiza zvočnih posnetkov glasu z uporabo metod umetne inteligence (npr. strojno učenje, procesiranje signalov itd.). Tovrsten način diagnostike je povsem varen, preprost, hiter in ne zahteva dragocenih namenskih naprav [8], vendar je to področje v primeru Parkinsonove bolezni še v razvoju. Večina raziskovalcev se namreč ukvarja le z doseganjem čim večje klasifikacijske točnosti [1, 7, 10, 11], pri tem pa zanemarjajo pomemben vidik analize, in sicer da bi skušali identificirati pomembne posnetke in pripadajoče glasovne attribute. Taka dognanja bi pripomogla k boljšemu razumevanju problematike in omogočila oblikovanje natančnejših testov.

V tem prispevku poročamo o testiranju uporabnosti analize glasu z metodami strojnega učenja za diagnosticiranje Parkinsonove bolezni. Opravljena študija temelji na zvočnih posnetkih 40 oseb (20 bolnikov s Parkinsonovo boleznijo) pridobljenih v raziskavi [10]. Na teh podatkih smo testirali pet različnih algoritmov strojnega učenja. Za izboljšanje rezultatov smo dodatno uporabili metodo za zmanjšanje razsežnosti in izboljšamo klasifikacijsko točnost za približno 5 %.

V nasprotju z večino sorodnega dela smo ocenili tudi pomembnost posameznih posnetkov in pripadajočih atributov. V ta namen smo uporabili metodo naključnih gozdov, saj ta dosega najvišjo točnost. Na ta način lahko ugotovimo, kateri atributi in posnetki vsebujejo več informacij o prisotnosti Parkinsonove bolezni.

Prispevek je organiziran na sledeči način. V drugem poglavju predstavimo podatke, v tretjem poglavju opišemo metodologijo. V četrtem in petem poglavju predstavimo rezultate in pridobljena dognanja. V zadnjem poglavju naredimo zaključek in orišemo nadaljnje delo.

2 PODATKI

Podatki so bili zbrani na Istanbulske fakulteti za medicino (Istanbul Faculty of Medicine, Istanbul University) leta 2014 [10]. Zbrali so zvočne posnetke 40 ljudi, 6 žensk ter 14 moških s Parkinsonovo boleznijo in 10 zdravih žensk ter 10 zdravih moških. Vsaka oseba je posnela 26 posnetkov, ki vključujejo samoglasnike, kratke stavke in besede. Natančneje, posnetki 1–3 predstavljajo trajajoče samoglasnike “a”, “o” in “u”, posnetki 4–13 predstavljajo števila od 1 do 10, posnetki 14–17 predstavljajo krajše stavke in posnetki 18–26 predstavljajo besede. Vsi posnetki so v turščini, posneti so bili z mikrofonom Trust MC-1500¹.

Vsaki osebi pripada 26 zvočnih posnetkov in vsakemu posnetku 26 linearnih ter frekvenčnih atributov, zgrajenih z uporabo programske opreme za akustično analizo Praat [2]. Vsi atributi so numerični in se jih običajno izračuna za analizo glasu [2, 10]. Povzeti so v Tabeli 1. Skupno je v množici podatkov 676 atributov in ciljni razred. Slednji je binaren in predstavlja prisotnost (pozitiven = 1) oziroma odsotnost (negativen = 0) Parkinsonove bolezni. Imena nekaterih atributov uporabljamo v angleščini, saj pripadajoči slovenski izrazi ne obstajajo.

3 METODOLOGIJA

Klasifikatorje smo gradili s petimi algoritmi za strojno učenje: odločitveno drevo (C4.5), naivni Bayes (NB), metoda najbližjih sosedov (k NN), metoda podpornih vektorjev (SVM) ter metoda naključnih gozdov (RF). Za vse navedene algoritme smo uporabili privzete vrednosti parametrov, saj ugaševanje ni signifikantno izboljšalo klasifikacijske točnosti.

Število atributov močno presega število primerkov, zato smo se odločili za uporabo metode zmanjševanja razsežnosti in s tem uspešno izboljšali klasifikacijsko točnost za 5 %. Za izbor atributov smo uporabili široko poznano metodo, imenovano rekurzivna odstranitev atributov (ang. *recursive feature elimination*, RFE) [4], ki temelji na vzratni odstranitvi nepomembnih atributov. Metoda RFE spada med metode po principu ovojnice (ang. *wrapper*) in smo jo uporabili v kombinaciji z zgoraj naštetimi algoritmi za strojno učenje. Končno število atributov, ki v RFE nastopa kot parameter, smo ocenili z 10-kratnim prečnim preverjanjem.

Za strojno učenje smo uporabili knjižnico caret [6], implementirano v programskem jeziku R [9].

4 REZULTATI

Za evalvacijo in izbor najboljšega algoritma smo uporabili pristop po metodi “izpusti enega” (ang. *leave one subject out*, LOSO). Najprej smo na učni množici z 10-kratnim prečnim preverjanjem ocenili končno število atributov, ki nastopa kot parameter metode RFE. Nato smo z ugašeno metodo RFE izbrali najboljše atribute in pripadajoči klasifikator. S slednjim smo klasificirali izpuščen primerek in opisan postopek ponovili za vse primerke.

¹<https://www.trust.com/en/product/14896-design-microphone-mc-1500>

Tabela 1: Glasovni atributi, uporabljeni za strojno učenje: frekvenčni, pulzni, amplitudni, glasovni ter harmonični.

Skupina	Atribut
Frekvenčni	Jitter (local)
	Jitter (local, absolute)
	Jitter (rap)
	Jitter (ppq5)
	Jitter (ddp)
Pulzni	Število glasovnih pulzov
	Število nihalnih dob
	Povprečna perioda
	Standardna deviacija period
Amplitudni	Shimmer (local)
	Shimmer (local, dB)
	Shimmer (apq3)
	Shimmer (apq5)
	Shimmer (apq11)
Glasovni	Delež nezvencenih časovnih oken
	Število lomljenj glasu
	Delež lomljenj glasu
Harmonični	Srednja vrednost višine glasu
	Povprečna višina glasu
	Standardna deviacija višine glasu
	Najvišja višina tona
	Najnižja višina tona
	Avtokorelacija tona
	Razmerje šum-harmonik
Razmerje harmonik-šum	

Tabela 2: Rezultati klasifikatorjev v obliki točnosti, senzitivnosti in specifičnosti. Najvišja vrednost posamezne metrike je odebeljena.

Algoritem	Točnost	Senzitivnost	Specifičnost
C4.5	0,63	0,65	0,60
NB	0,63	0,80	0,45
k NN	0,48	0,55	0,40
SVM	0,68	0,70	0,65
RF	0,73	0,75	0,70

Tabela 3: Matrika zamenjav za klasifikator, zgrajen z metodo RF.

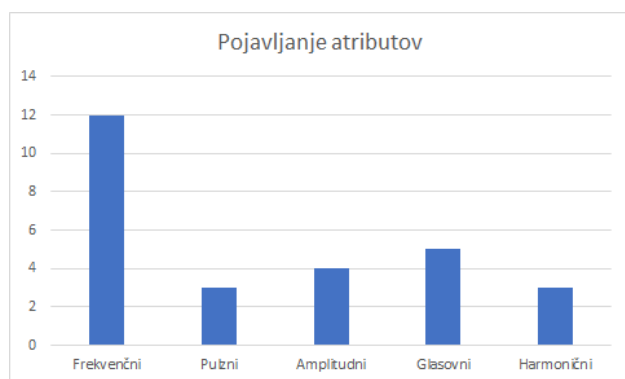
Napoved / Pravi	Negativen (0)	Pozitiven (1)
Negativen (0)	14	5
Pozitiven (1)	6	15

V Tabeli 2 so prikazani rezultati v obliki povprečne točnosti, povprečne senzitivnosti in povprečne specifičnosti. Vidimo, da je najbolj točen klasifikator, zgrajen z metodo RF, najmanj točen pa z metodo k NN. Najvišjo senzitivnost je dosegel klasifikator, zgrajen z metodo NB, specifičnost pa klasifikator, zgrajen z metodo RF. V Tabeli 3 so prikazani rezultati za klasifikator, zgrajen z metodo RF v obliki matrike zamenjav. Klasifikator je pravilno klasificiral 29 primerkov, zmotil pa se je v 11 primerih.

Zanimala nas je pomembnost posameznih posnetkov in pripadajočih atributov. V ta namen smo postopek izbora atributov ponovili za RF, a tokrat na celotnih podatkih brez izpusta primerkov. Pomembnost izbranih posnetkov in atributov smo izračunali s postopkom, imenovanim permutacijska pomembnost (ang. *permutation importance*), ki ga lahko neposredno vključimo v metodo RF [3]. Za vsako drevo posebej izračunamo točnost na izpuščenih primerkih (naključno izpuščenih za gradnjo drevesa). Nato ponovimo izračun točnosti po permutaciji določenega atributa. Pomembnost tega atributa je povprečje razlik v točnosti pred in po njegovi permutaciji. Pri tem poudarimo, da pri metodi RF ni težav s koreliranimi atributi, saj postopek uporabimo na posameznem drevesu, ki je po načinu izgradnje nekoreliran.

Na ta način izberemo 27 izmed 676 atributov. Med njimi se najpogosteje pojavljajo frekvenčni atributi (Slika 1), medtem ko so ostale skupine atributov podobno zastopane. Med posnetki se najpogosteje pojavljajo števila, nato kratki stavki. Najslabše zastopani so trajajoči samoglasniki (Slika 2).

Slika 3 in Slika 4 predstavljata zaporedoma pomembnost izbranih atributov (agregirano čez posnetke) in pomembnost posnetkov (agregirano čez attribute) za metodo RF. Atributi in posnetki so razvrščeni od manj pomembnih do bolj pomembnih. Iz rezultatov je razvidno, da so za metodo RF najpomembnejši frekvenčni atributi. Najmanj pomembni pa so harmonični atributi in atributi, izpeljani iz tona glasu. Najpomembnejši posnetek je število "4". Opazimo, da števila in kratki stavki vsebujejo več informacij od ostalih posnetkov.



Slika 1: Število izbranih atributov za posamezne skupine po uporabi metode RFE v kombinaciji z metodo RF.



Slika 2: Število izbranih posnetkov za posamezne skupine po uporabi metode RFE v kombinaciji z metodo RF.

5 DISKUSIJA

Podobno kot sorodne raziskave [1, 7, 10, 11] tudi naši rezultati nakazujejo na povezavo med glasovnimi atributi in prisotnostjo Parkinsonove bolezni. Najbolj točen klasifikator zgradimo z uporabo metode RF, s katerim dosežemo 73 % točnost. Za primerjavo nekatera sorodna dela poročajo o točnosti okoli 80 %.

Pri tem so najpomembnejši in pogosti frekvenčni atributi (Slika 1 in Slika 3). Sklepamo, da zaradi karakteristične deviacije frekvence glasu pri Parkinsonovi bolezni. Med posnetki izstopajo števila in kratki stavki (Slika 2 in Slika 4). O prisotnosti bolezni nam več povedo zahtevni ter daljši posnetki.

Kljub temu je tak način diagnoze nezadosten. Najbolj točna metoda zgreši 25 % bolnikov, kar je za medicinsko prakso nesprejemljivo [13]. Pri tem moramo poudariti, da smo imeli opravka z omejenim številom primerkov (posnetih je bilo le 40 oseb). V primeru, da bi zbrali več zvočnih posnetkov obolelih in zdravih oseb, bi lahko klasifikator izboljšali z uporabo naprednejših metod strojnega učenja, ki jih na tako majnem številu primerkov ni bilo moč uporabiti.

Morda ne bo nikoli moč stoodstotno določiti prisotnost Parkinsonove bolezni iz analize glasu z uporabo metod strojnega učenja, vendar bi tovrstne metode lahko uporabili bodisi komplementarno za nadgradnjo obstoječih metod bodisi kot presejalni test. Pri tem poudarimo, da je analiza glasu poceni in za bolnika povsem nemoteča ter varna preiskava.

6 ZAKLJUČEK

V prispevku smo z metodami strojnega učenja primerjali zvočne posnetke zdravih oseb in bolnikov s Parkinsonovo boleznijo. Namen študije je bil preveriti, ali lahko iz analize glasu sklepamo o prisotnosti Parkinsonove bolezni in ali je možno zgraditi klasifikator za uporabo v praksi. Dodatno smo tudi ocenili pomembnost posameznih posnetkov in pripadajočih glasovnih atributov.

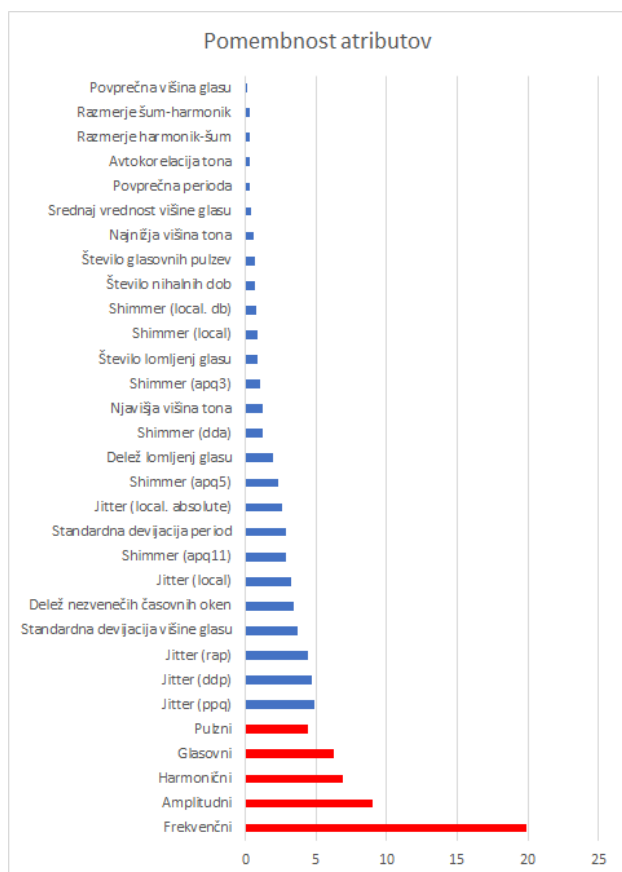
Rezultati nakazujejo, da pri bolnikih s Parkinsonovo boleznijo pride do poslabšanja zvočne artikulacije, saj smo s klasifikatorjem, zgrajenim z metodo naključnih gozdov, uspešno zaznali 73 % bolnikov. Ne glede na to klasifikator še ni primeren za uporabo v praksi, saj je njegova točnost prenizka. Sedanji klasifikator lahko uporabimo kot komplementarni test že obstoječim. Za najpomembnejše zvočne posnetke se izkažejo števila in kratki stavki. Pri tem so najmanj pomembni trajajoči samoglasniki in besede. Med atributi izstopajo frekvenčni in amplitudni.

Trenutno raziskujemo možnost, da bi zbrali več sorodnih zvočnih posnetkov. Na ta način bi lahko uporabili kompleksnejše metode, ki omogočajo odkrivanje zagonetnih zakonitosti, ki jih na tako majhnem naboru primerkov ni bilo mogoče odkriti.

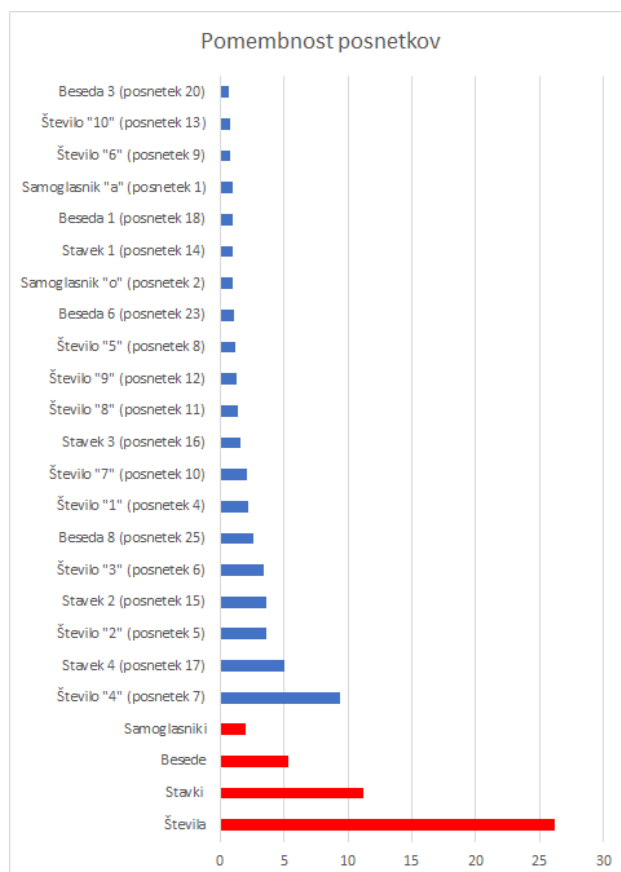
Naš dolgoročni cilj je izgradnja klasifikatorja, ki bi uspešno identificiral večino bolnikov tudi za ceno nekoliko nižje točnosti (nekateri zdrave osebe bi klasificiral za bolne). Klasifikator bi lahko uporabili kot presejalni test in na ta način olajšali sedanjo diagnostiko Parkinsonove bolezni. Poskusili bomo tudi razbrati, zakaj so ravno posnetki števil vsebovali več informacij o prisotnosti Parkinsonove bolezni, in z dobljenim znanjem skušali predlagati celovitejši nabor izrazov, besed in fonemov.

ZAHVALA

Avtorji se zahvaljujejo gospe Ireni Hočvar Boltežar za razlago glasovnih atributov in slovenske prevode. A. Vodopija se dodatno zahvaljuje finančni podpori Javne agencije za raziskovalno dejavnost Republike Slovenije (program usposabljanja mladega raziskovalca).



Slika 3: Pomembnost izbranih atributov za klasifikator, zgrajen z metodo RF. Pomembnost posamezne skupine je agregirana pomembnost pripadajočih atributov.



Slika 4: Pomembnost izbranih posnetkov za klasifikator, zgrajen z metodo RF. Pomembnost posamezne skupine je agregirana pomembnost pripadajočih posnetkov.

LITERATURA

- [1] I. Bhattacharya in M. P. S. Bhatia. 2010. SVM classification to distinguish parkinson disease patients. V *Proceedings of the 1st Amrita ACM-W Celebration on Women in Computing in India*. ACM, New York, NY, USA, 1–6. doi: 10.1145/1858378.1858392.
- [2] P. Boersma. 2001. Praat, a system for doing phonetics by computer. *Glott International*, 5, 9/10, 341–345.
- [3] L. Breiman. 2001. Random forests. *Machine Learning*, 45, 1, 5–32. doi: 10.1023/A:1010933404324.
- [4] I. Guyon, J. Weston, S. Barnhill in V. Vapnik. 2002. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46, 1, 389–422. doi: 10.1023/A:1012487302797.
- [5] I. Hočvar Boltežar. 2013. *Fiziologija in patologija glasu ter izbrana poglavja iz patologije govora*. Pedagoška fakulteta. <http://www.biblos.si/lib/book/9789612531416>.
- [6] M. Kuhn. 2008. Building predictive models in R using the caret package. *Journal of Statistical Software, Articles*, 28, 5, 1–26. doi: 10.18637/jss.v028.i05.
- [7] M. A. Little, P. E. McSharry, E. J. Hunter, J. Spielman in L. O. Ramig. 2009. Suitability of dysphonia measurements for telemonitoring of parkinson's disease. *IEEE Transactions on Biomedical Engineering*, 56, 4, 1015–1022. doi: 10.1109/TBME.2008.2005954.
- [8] M. A. Little, P. E. McSharry, S. Roberts, D. Costello in I. Moroz. 2007. Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection. *Nature Precedings*. doi: 10.1038/npre.2007.326.1.
- [9] R Core Team. 2013. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. <http://www.R-project.org/>.
- [10] B. E. Sakar, M. E. Isenkul, C. O. Sakar, A. Sertbas, F. Gurgun, S. Delil, H. Apaydin in O. Kursun. 2013. Collection and analysis of a parkinson speech dataset with multiple types of sound recordings. *IEEE Journal of Biomedical and Health Informatics*, 17, 4, 828–834. doi: 10.1109/JBHI.2013.2245674.
- [11] C. O. Sakar in O. Kursun. 2010. Tediagnosis of parkinson's disease using measurements of dysphonia. *Journal of Medical Systems*, 34, 4, 591–599. doi: 10.1007/s10916-009-9272-y.
- [12] C. Silva. 2018. Speech analysis may help diagnose parkinson's and at earlier stage, study says. *Parkinson's News Today*. (2018). <https://parkinsonsnewstoday.com/2018/02/05/speech-analysis-can-help-detect-parkinsons-in-early-stages-study-says/>.
- [13] E. Tolosa, G. Wenning in W. Poewe. 2006. The diagnosis of parkinson's disease. *The Lancet Neurology*, 5, 1, 75–86. doi: 10.1016/S1474-4422(05)70285-4.

STRAW Application for Collecting Context Data and Ecological Momentary Assessment

Junoš Lukan
Jožef Stefan Institute
Jožef Stefan International
Postgraduate School
Jamova cesta 39
Ljubljana, Slovenia
junos.lukan@ijs.si

Marko Katrašnik
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia
marko.katrasnik@gmail.com

Larissa Bolliger
Department of Public Health
Ghent University
Ghent, Belgium
larissa.bolliger@ugent.be

Els Clays
Department of Public Health
Ghent University
Ghent, Belgium
els.clays@ugent.be

Mitja Luštrek
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia
mitja.lustrek@ijs.si

ABSTRACT

To study stress at the workplace and relate it to user context and self-reports, we developed an application based on the AWARE framework, a mobile instrumentation toolkit. The application serves two purposes: of passively collecting data about user's environment and offering questionnaires as means of ecological momentary assessment. We implemented methods to import the questionnaires into the phone's database and trigger them at the right times. We also considered privacy implications of collecting such data and took additional measures to conceal the identity of our study's participants wherever we evaluated it was under the risk of exposure. Finally, we had to establish a server application to handle receiving and storage of collected data and implemented a rudimentary login process to additionally secure our servers.

KEYWORDS

context detection, application development, privacy, ecological momentary assessment

1 APPLICATION OVERVIEW

The best machine learning models for stress detection and affect recognition are multimodal [1, 17]. Combining data from different modalities is especially effective, such as using physiological, behavioural or contextual, and psychological (self-reported) data. Collecting such data in the real-world setting presents a challenge, however.

In the project called *Stress at work* (STRAW), the main objective is to analyse the relationship between psychosocial stress experiences in the workplace, work activities and events, and peripheral physiology. To facilitate integration of various data sources, an application was designed to run continuously and monitor their environment and specific phone-related events.

The application's purpose is two-fold. The primary mode of operation is silent and continuous: the user context (such as their

phone use and location) is monitored without user intervention or interaction. The second mode of operation are prompts or questions for the user, where some information about the context and the participant's mental state is gathered by asking for it explicitly.

As a starting point for writing the STRAW application, we used AWARE, a mobile instrumentation toolkit which had the initial purpose of inferring users' context [5]. It enables logging of data as reported by the phone's operating system and a wide variety of hardware sensors. At several points, this toolkit was adapted to better suit our needs, and additional capabilities were added on top of it.

We also developed two modular functionalities of the application: Bluetooth integration with an Empatica E4 wristband [23] to enable simultaneous collection of physiological data and voice detection and speaker diarization capabilities [15]. We already reported on these developments elsewhere, whereas in this paper, we give an overview of the app's capabilities.

1.1 Data Types

An important aspect of the STRAW application are prompts, called EMAs. The users can be prompted to make a diary entry at a specific time which is called Experience Sampling Method [ESM; 3] or, more broadly (when data other than experience are noted), Ecological Momentary Assessment [EMA; 20]. Diary methods increase the reliability of collected self-reports as they are less prone to recall bias [14].

EMAs are the main mode of user interaction in the STRAW application. The content of specific questions is beyond the scope of this paper, but in general, the questions are based on existing psychological questionnaires measuring stressors, stress, and related responses. The implementation of EMAs is described in Section 2.

In addition to this, we selected a subset of data that might help us determine users' context. Below is a list of sensors that are used in the STRAW application together with the description of data they collect. Data availability from some of these sensors is dependent on phone's hardware and the version of operating system.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

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

© 2020 Copyright held by the owner/author(s).

- **ACCELERATION:** There are several sources (i.e. virtual sensors) of acceleration data in a smartphone. Accelerometers measure acceleration magnitude in various directions and report either linear acceleration (without gravity effects), gravity, or combined acceleration. This is used further in Google’s activity recognition API [10].
- **BAROMETER:** Ambient air pressure.
- **LIGHT:** Luminance of the ambient light captured by the light sensor.
- **TEMPERATURE:** Temperature of the phone’s hardware sensor.
- **BLUETOOTH:** This sensor logs surrounding Bluetooth-enabled and visible devices, specifically their hashed MAC addresses, and received signal strength indicator (RSSI) in decibels.
- **LOCATION:** Device’s current location (latitude, longitude, and altitude, which are masked as described in Section 3) and its velocity (speed and bearing). This uses various methods, such as GPS and known Wi-Fis in vicinity resulting in different degrees of accuracy. Location category is also acquired with Foursquare API.
- **NETWORK:** Network availability (e.g. none or aeroplane mode, Wi-Fi, Bluetooth, GPS, mobile) and traffic data (received and sent packets and bytes over either Wi-Fi or mobile data).
- **PROXIMITY:** Uses the sensor by the device’s display to detect nearby objects. It can either be a binary indicator of an object’s presence or the distance to the object.
- **TIMEZONE:** Device’s current time zone.
- **WI-FI:** Logs of surrounding Wi-Fi access points, specifically their hashed MAC addresses, received signal strength indicator (RSSI) in decibels, security protocols, and band frequency. The information on the currently connected access point is also included.
- **APPLICATIONS:** This includes the category of the application currently in use (i.e. running in the foreground) and data related to notifications that any application sends. Notification header text (but not content), the category of the application that triggered the notification and delivery modes (such as sound, vibration and LED light) are logged.
- **BATTERY:** Battery information, such as current battery percentage level, voltage, and temperature, and its health, as well as power-related events, such as charging and discharging times are monitored.
- **COMMUNICATION:** Information about calls and messages sent or received by the user. This includes the call or message type (i.e. incoming, outgoing, or missed), length of the call session, and trace, a SHA-1 encrypted phone number that was contacted. The phone numbers themselves or the contents of messages and calls are not logged.
- **PROCESSOR:** Processor load in CPU ticks and the percentage of load dedicated to user and system processes or idle load.
- **SCREEN:** Screen status: turned on or off and locked or unlocked.
- **VOICE ACTIVITY:** A classifier, trained using Weka [7]. The features are calculated using openSMILE [4] and the output is an indicator of human voice activity [15].

The data described in the list above are collected automatically and continuously. The application is run as a foreground service, which means that the data collection continues even while the application is not actively used (i.e. it is minimized). Despite this, there exists software that is specific to the operating system

version and phone manufacturer which tries to close applications for energy efficiency. We attempted to whitelist this application in the most common battery-saving software.

2 ECOLOGICAL MOMENTARY ASSESSMENT

As mentioned, one of the main functions of the STRAW application is to collect users’ answers to questionnaires. AWARE already implements a ‘sensor’ for experience sampling method, which shows DialogFragments as the one in Figure 1, but it was too rudimentary for our study protocol. The main upgrades we had to make were the mechanism of triggering EMAs and management of the database of available questions (items) to include in the questionnaires.

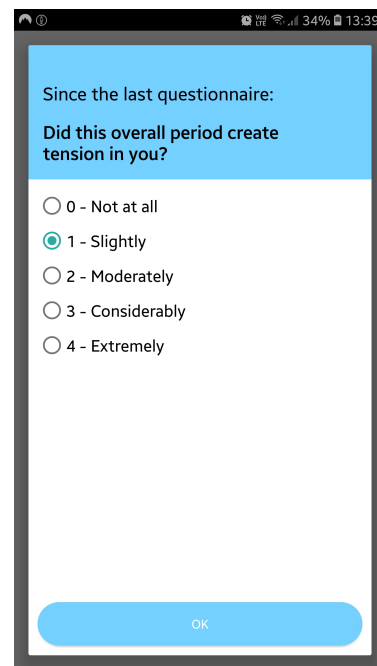


Figure 1: An example of an ecological momentary assessment prompt.

2.1 EMA Triggering

Originally, AWARE provides a couple of ways to trigger EMAs: at a specific time, by a certain context (i.e. taking into account values from other sensors) or on demand (manually). In our study, time is the most important trigger of EMAs, but we needed finer control.

The EMAs in our studies are divided into three types: a) morning EMAs with questions about sleep quality, b) work-hour EMAs with questions about momentary affect, job characteristics, work activities, and similar, and c) evening EMAs with questions about the whole workday and after-work activities. The first EMA is triggered in the first hour after the start of the workday as set by the user. The rest of the EMAs during work hours trigger approximately every 90 minutes, but not closer than 30 min apart. The time is dependent on the last answered EMA rather than set in advance, and additional reminders are scheduled in the case of user inactivity. The final EMA of the day is triggered in the evening at a time set by the user.

Each of these types of EMA is implemented as a separate `IntentService` [11] and handled by a `JobScheduler` [18]. This enabled us to enforce the requirements outlined above such as setting the minimum latency with which the job can start and making use of periodic jobs.

2.2 Question Database

In the original AWARE implementation, questions are queued into a questionnaire directly in the code of the application by using their custom `ESMFactory` class. For our study, we use a pool of more than 200 questions per language from which a subset is sampled for every EMA. We therefore needed a more systematic way of storing them within the application.

To ease the insertion of individual items, we prepared a spreadsheet template which is meant to be human-readable and filled out manually. Individual items from this spreadsheet are later converted into JavaScript Object Notation (JSON) and stored in an SQLite database [13] in phone's internal storage. This implementation enabled us to adapt the content of EMAs without touching the source code of the application. It also simplified the final selection of questions, such as selecting one language (English, Dutch, or Slovenian) and grammatical gender.

3 PRIVACY ENHANCEMENTS

The data collected by the STRAW application have different degrees of risk to the users' privacy. Their privacy would be threatened if an outsider gained unauthorized access to the data. These possible external threats are considered in Section 5.

Even when the data are safely communicated and stored, however, an involuntary exposure of users' identity might still be possible. Assuming the data are well protected from unauthorized external access, these risks will in turn be treated as internal in this section.

Some of the data collected by the STRAW application are personal data, so even when storing them securely and after pseudonymization, some risk of a privacy breach remains. Since AWARE is widely used in scientific studies it already implements some privacy enhancing mechanisms. We performed a thorough application vulnerability analysis and identified several further threats to privacy that we wished to address. While the data are safely communicated and stored, an involuntary exposure of users' identity might still be possible. The types of data that deserve special attention are applications, communication, location, and voice activity.

As mentioned in Section 1, the notifications that other applications send are monitored in the STRAW application. The content of the notification, such as that of an instant messaging application or calendar notification, is never actually stored. We deemed even the application names to be sensitive, so we chose to only save application categories. This process is further described in Section 4.

The content of calls or messages is never logged, but the phone numbers tied to them can be. Since we wanted to keep track of recurring contact with the same person, but not reveal their real phone number, we decided to encrypt them using the SHA-1 algorithm. While it would be possible to decrypt a phone number by a brute-force attack, the AWARE implementation offers the option of adding a salt. Thus by using the username (further described in Section 5) as a salt, the phone numbers are sufficiently protected from inadvertent disclosure risk, while the hashed value is retained even across different application installations.

The MAC addresses of detected WiFi and Bluetooth devices are hashed in the same way.

The location data in their raw form are highly revealing of a user's identity [2]. Instead of storing the actual geographic coordinates provided by this sensor, the Foursquare Places API [6] is used to extract the category (venue) of a location. This API enables saving general categories such as 'bookstore' or 'gas station' near the user's location. But since we wanted to keep the option to analyse users' movements, we also implemented a transformation of coordinates. We converted longitude and latitude into spherical coordinates, applied a stochastic rotation (but constant within a specific user) and converted these back to transformed longitude and latitude. This enabled us to keep the distances between the locations faithful to original data, but transformed to another place on Earth.

As described in our previous work [15], voice activity recognition is performed on the phone in its entirety. This means that raw audio recordings can be discarded immediately after processing and only the calculated features are saved to the database. Alternatively, only the final binary prediction of human voice presence can be retained, but this makes any post-hoc analysis (such as speaker diarization) impossible.

4 SERVER APPLICATION

For the purpose of storing the data on a server, a Python application was implemented in Flask [21], which accepts the data in a JSON format and saves it in a PostgreSQL [22] database. In addition to receiving the data and managing credentials (as described in Section 5), it also performs a couple of additional functions.

As mentioned in Section 3, instead of saving application names we only log their category as classified in Google Play Store. To reduce the number of queries, we implemented this as a part of the server application. As part of the upload process, the application name is received in plain text, but only retained until query returns its category. After that, the application name is hashed to enable comparisons with later records and the name in plain text is discarded. In this way, we could build a database of application name hashes and their corresponding categories on the server, while not keeping a record of what applications individual users use.

The server application also provides a simple UI for administrators, where some metadata about the data collection itself are shown in forms of tables and charts. We can access data on last upload, number of days of participation, and number of data points for each individual user. This enables us to detect any problems with data collection and troubleshoot them early.

5 CLIENT-SERVER COMMUNICATION AND LOGIN

The STRAW application and other sensing applications are not special in the degree they could be subject to external attacks [2]. An attacker might want to expose identity of a user or try to reveal their personal data such as location. There are three points of entry for an external attacker: local storage, transmission of data, and the servers.

While the data reside on the device they are saved locally in the phone's storage. According to Android's documentation, this database is exclusive to the STRAW application [9]:

Other applications cannot access files stored within internal storage. This makes internal storage a good

place for application data that other applications shouldn't access.

Additionally, once the data are transmitted to the server, the local database is periodically deleted. This reduces the privacy risk of the database being exposed, while also decreasing the local storage requirements.

It is therefore the transmission of data where we had to secure the data. They are transmitted over encrypted HTTPS connection, which eliminates the risk of exposure during this part of communication. The data are received by an application server residing at Jožef Stefan Institute (JSI), with a dedicated port listening for incoming transmissions.

The application server communicates with another, database server, also residing at JSI. This second server can only be accessed from within the JSI local area network. The database itself is also protected with a password and the user accessing it via the application server does not have administrator privileges.

Since the STRAW application is a part of a wider study, it is disseminated to recruited participants only. In addition to the data from this application, other data are collected, such as responses to questionnaires in baseline screening and physiological data from wristbands. It was therefore necessary that the data can be linked back to an individual in order to join the data from various sources. We developed a login method to enable this.

Using OkHttp [19] client-side and Flask-HTTPAuth [12] server-side, we implemented basic access authentication and token authentication [16]. The login credentials are disseminated to registered participants in our study and are input upon the installation of the STRAW application. This serves multiple purposes: by requiring login, we only accept data from actual participants of our study, while we can also use the assigned username to pseudoanonymously link data from various sources.

6 CONCLUSION

The application used in the STRAW project serves a dual purpose: to collect users' answers to questionnaires and passively collect data about their environment and phone usage. While the application was tailored to requirements of our study, this paper outlined the main issues and possible solutions when developing an application for research purposes.

The AWARE framework provided a solid foundation and especially eased sensor data collection, there are additional challenges that researchers need to face when trying to use an application like this in a scientific study. The data gathered using this application will help us develop improved models of stress recognition [8], which will help us integrate physiological data with more detailed contextual data and more reliable self-reports.

ACKNOWLEDGMENTS

The authors acknowledge the STRAW project was financially supported by the Slovenian Research Agency (ARRS, project ID N2-0081) and by the Research Foundation – Flanders, Belgium (FWO, project no. G.0318.18N).

REFERENCES

- [1] Ane Alberdi, Asier Aztiria and Adrian Basarab. 2016. Towards an automatic early stress recognition system for office environments based on multimodal measurements. A review. *Journal of Biomedical Informatics*, 59, (February 2016), 49–75. doi: 10.1016/j.jbi.2015.11.007.
- [2] Delphine Christin. 2016. Privacy in mobile participatory sensing. Current trends and future challenges. *Journal of Systems and Software*, 116, 57–68. doi: 10.1016/j.jss.2015.03.067.
- [3] Mihaly Csikszentmihalyi, Reed Larson and Suzanne Prescott. 1977. The ecology of adolescent activity and experience. *Journal of Youth and Adolescence*, 6, 3, (September 1977), 281–294. doi: 10.1007/bf02138940.
- [4] Florian Eyben, Felix Wengler, Florian Gross and Björn Schuller. 2013. Recent developments in openSMILE, the Munich open-source multimedia feature extractor. In *Proceedings of the 21st ACM international conference on Multimedia - MM '13*. ACM Press. doi: 10.1145/2502081.2502224.
- [5] Denzil Ferreira, Vassilis Kostakos and Anind K. Dey. 2015. AWARE: Mobile context instrumentation framework. *Frontiers in ICT*, 2, 6, 1–9. ISSN: 2297-198X. doi: 10.3389/fict.2015.00006. <https://www.frontiersin.org/article/10.3389/fict.2015.00006>.
- [6] Foursquare. [n. d.] Places SDK. Venue search. Retrieved 26/08/2020 from <https://developer.foursquare.com/docs/api-reference/venues/search/>.
- [7] Eibe Frank, Mark A. Hall and Ian H. Witten. 2016. *The WEKA workbench*. (4th edition). Morgan Kaufmann.
- [8] Martin Gjoreski, Mitja Luštrek, Matjaž Gams and Hristijan Gjoreski. 2017. Monitoring stress with a wrist device using context. *Journal of Biomedical Informatics*, 73, 159–170. ISSN: 1532-0464. doi: 10.1016/j.jbi.2017.08.006.
- [9] Google. [n. d.] Access app-specific files. Access from internal storage. Retrieved 26/08/2020 from <https://developer.android.com/training/data-storage/app-specific>.
- [10] Google. [n. d.] Adapt your app by understanding what users are doing. Retrieved 26/08/2020 from <https://developers.google.com/location-context/activity-recognition>.
- [11] Google. [n. d.] IntentService. Retrieved 26/08/2020 from <https://developer.android.com/reference/android/app/IntentService.html>.
- [12] Miguel Grinberg. [n. d.] Flask-HTTPAuth. Retrieved 26/08/2020 from <https://flask-httpauth.readthedocs.io/en/latest/>.
- [13] D. Richard Hipp, Dan Kennedy and Joe Mistachkin. 2019. SQLite. Computer software. (2019). <https://sqlite.org/index.html>.
- [14] Gillian H. Ice and Gary D. James, editors. 2006. *Measuring emotional and behavioral response. General principles. Measuring Stress in Humans. A Practical Guide for the Field*. Part II – Measuring stress responses. Cambridge University Press, Cambridge, UK, (December 2006). Chapter 3, 60–93. ISBN: 978-0-521-84479-6.
- [15] Marko Katrašnik, Junoš Lukan, Mitja Luštrek and Vitomir Štruc. 2019. Razvoj postopka diarizacije govorcev z algoritmi strojnega učenja. In *Proceedings of the 22nd International Multiconference INFORMATION SOCIETY – IS 2019*. Slovenian Conference on Artificial Intelligence. Mitja Luštrek, Matjaž Gams and Rok Piltaver, editors. Volume A, 57–60. <https://is.ijs.si/archive/proceedings/2018/files/Zbornik%20-%20A.pdf>.
- [16] Chris Schmidt. 2001. Token based authentication. In *Accepted papers for FOAF-Galway*. 1st Workshop on Friend of a Friend, Social Networking and the Semantic Web. https://www.w3.org/2001/sw/Europe/events/foaf-galway/papers/fp/token_based_authentication/.

- [17] Philip Schmidt, Attila Reiss, Robert Duerichen and Kristof Van Laerhoven. Wearable affect and stress recognition: a review. (21st November 2018).
- [18] Joanna Smith. 2016. Scheduling jobs like a pro with JobScheduler. <https://medium.com/google-developers/scheduling-jobs-like-a-pro-with-jobscheduler-286ef8510129>.
- [19] Square, Inc. 2019. OkHttp. Computer software. (2019). <https://square.github.io/okhttp/>.
- [20] Arthur A. Stone and Saul Shiffman. 1994. Ecological momentary assessment (EMA) in behavioral medicine. *Annals of Behavioral Medicine*, 16, 3, 199–202. DOI: 10.1093/abm/16.3.199.
- [21] The Pallets team. 2010. Flask. Computer software. (2010). <http://flask.pocoo.org/>.
- [22] The PostgreSQL Global Development Group. 2019. *PostgreSQL 11.3 Documentation*. Version 11.3.
- [23] Marija Trajanoska, Marko Katrašnik, Junoš Lukan, Martin Gjoreski, Hristijan Gjoreski and Mitja Luštrek. 2018. Context-aware stress detection in the aware framework. In *Proceedings of the 21st International Multiconference INFORMATION SOCIETY – IS 2018*. Slovenian Conference on Artificial Intelligence. Mitja Luštrek, Rok Piltaver and Matjaž Gams, editors. Volume A, 25–28. <https://is.ijs.si/archive/proceedings/2018/files/Zbornik%20-%20A.pdf>.

URBANITE H2020 Project

Algorithms and Simulation Techniques for Decision - Makers

Alina Machidon
alina.machidon@ijs.si
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia

Maj Smerkol
maj.smerkol@ijs.si
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia

Matjaž Gams
matjaz.gams@ijs.si
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia

ABSTRACT

URBANITE (Supporting the decision-making in URBAN transformation with the use of dISruptive TEchnologies) is a H2020 project with the goal to provide an ecosystem model that articulates the expectations, trust and attitude from civil servants, citizens and other stakeholders in the use of disruptive technologies. This model will be supported with the provision of a data management platform and algorithms for data – driven decision – making in the field of urban transformation. One of the main output of the project will be a Decision-Support System including (AI based) predictive algorithms and simulation models for mobility that support the decision-making process by analyzing the current situation, the trends that occurred in a certain time frame and allowing to predict future situations, when changing one or more variables. URBANITE will analyze the impact, trust and attitudes of civil servants, citizens and other stakeholders with respect to the integration of disruptive technologies such as Artificial Intelligence (AI), Decision Support Systems (DSS), big data analytics and predictive algorithms in a data-driven decision-making process. The results of the project will be validated in four real use cases: Amsterdam, Bilbao, Helsinki and Messina. This paper overviews the current state of the project's progress.

KEYWORDS

AI, Big Data, DSS, disruptive technologies, URBANITE project

1 INTRODUCTION

In recent times, the cities and urban environments are facing a revolution in urban mobility, bringing up unforeseen consequences that public administrations need to manage. It is in this new context that public administrations and policy makers need means to help them understand this new scenario, supporting them in making policy-related decisions and predicting eventualities. The traditional technological solutions are no longer valid for this situation and therefore, disruptive technologies such as big data analytics, predictive algorithms as well as decision support systems profiting from artificial intelligence techniques to support policy – makers come into place.

The main technical objective of the URBANITE project is the development of advanced AI algorithms for analysis of big data on mobility. The developed methods and tools will provide substantial support for policy-makers to tackle complex policy problems on the mobility domain and will enable their validation

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

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

© 2020 Copyright held by the owner/author(s).

on case-specific models. The goal of the activities will be to implement novel tools and services in order to enable policy-makers to use advanced data analysis and machine learning methods during the design of novel policies for a specific city

URBANITE will allow the analysis of the traffic flows that are currently happening and have happened up until that moment. In addition to the visualization of the traffic, usage of economy sharing vehicles and other aspects, URBANITE will analyse which are the bottlenecks and critical points, based on a set of parameters to be determined by the civil servants. Due to the fact that historic data is stored, trends can be determined by URBANITE by big data algorithms. These trend analyses can entail the understanding of, for instance, the use of a certain transportation system (e.g. bikes) in a certain neighbourhood of the municipality, or the peak hours in which a street is blocked. URBANITE will also provide means to simulate the effect of different situations such as opening a pedestrian street at certain times, location of electric charging stations, or bike sharing points through the implementation of artificial intelligence algorithms. To achieve that, URBANITE will build first generic models from the data across all the cities and then provide adaptation mechanisms to apply these models to the different use cases. From the data available, URBANITE will extract and formalize knowledge and then, through a combination of classification, regression, clustering, and frequent pattern mining algorithms, conclude into some decisions and actionable models that will enable city policy-makers to simulate and assess the outcomes and implications of new policies.

2 SYSTEM'S ARCHITECTURE

The URBANITE project will combine various data sources, algorithms, libraries and tools that provide the best solutions to the scope of the project. The technical "core" of the project has to fulfill the following objectives:

- Deploy tools for big data exploration with the active involvement of policy-makers.
- Design methods for the detection of important events that need to be addressed.

In order to provide the desired functionalities, several state-of-the-art technologies are currently examined and tested in order to be adapted, customized and integrated into the platform. A simplified preliminary architecture is presented in Figure 1.

2.1 Data Analysis Module

One of the first tasks involves the development of various methods for exploratory data analysis and user interaction. Multimodal methods, tools and services for big data on urban mobility will be implemented that will provide exploratory analysis capabilities and enable the policy-makers to actively search for causal relations in the data will be provided by the platform.

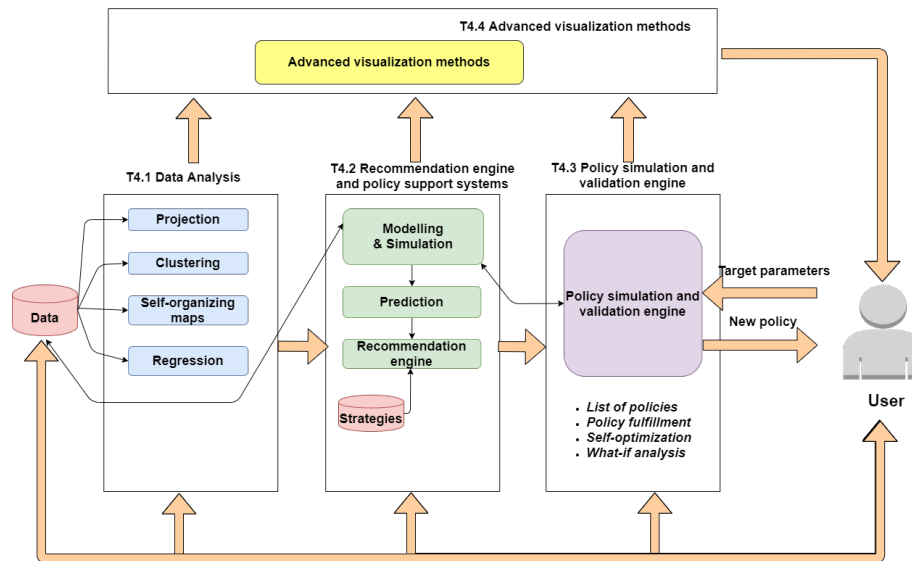


Figure 1: High Level Architecture of the URBANITE Platform.

The methods to be included in the platform can be segmented in four main groups:

- clustering, where the main goal is to reduce the amount of data by grouping together similar instances. The implemented method will provide mechanisms to group instances based on GIS data or any subset of attributes that users will define. For example, platform users might choose to cluster all instances based on the type of transportation used (shared bikes, electric cars, etc.)
- projection methods that will be used to reduce the dimensionality of the data items. The goal of these methods is to represent the data in a lower dimensional space in such a way that the key relations of the data structures are preserved. The results of the methods can be used to more clearly visualize the data or use the transformed data in the next rounds of analysis
- self-organizing map involves the use of a type of artificial neural network, trained in an unsupervised manner. The method can at the same time reduce the amount of data (similar to clustering) and nonlinearly projects the data into lower dimensionalities
- prediction/regression methods, or classification models, that will allow to exploit the data

2.2 Recommendation Engine

Recommendation engines (also known as recommender systems) are information filtering systems that deal with the problem of information overload [6] by filtering key information "chunks" out of large amount of dynamically generated information according to user's preferences, interest, or observed behavior about item [8][5]. Recommendation engines have the ability to predict whether a particular user would prefer an item or not based on the user's profile [5]. Recommendation engine is defined as a decision making strategy for users under complex information environments [4]. Recently, various approaches for building recommendation engines were developed, based on either collaborative filtering, content-based filtering or hybrid filtering [12], [11], [9].

The URBANITE recommendation engine will identify and predict important or problematic events related to mobility and will provide suggestions to tackle the issue. The policy support system will provide support to the policy-makers for identifying possible policies that tackle events based on specific criteria. The inputs will have to be aggregated for effective decision-making using hierarchical multi-criteria decision models.

2.3 Policy Simulation and Validation Engine

Simulation transparency is a vital feature of the decision making process when quantitative computer tools are used to justify some strategies [10]. Simulation predictions can play a catalytic role in the development of public policies, in the elaboration of safety procedures, and in establishing legal liability. Hence, given the impact that modelling and simulation predictions are known to have, the credibility of the computational results is of crucial importance to engineering designers and managers but also to public servants, and to all citizens affected by the decisions that are based on these predictions [10].

To create trust and increase the model's credibility and the simulation results delivered, it is crucial to deal with a validation strategy in which non-simulation-trained end-users could feel comfortable and trust the simulation model [10].

In the URBANITE project, the policy simulation and validation module will provide methods and tools to simulate the efficiency of specific policies in the target domain. Given a new policy, urban mobility model and the target parameters, the system can evaluate the performance of the new policy based on the observed parameters. The implementation of credible traffic simulations for the entire city has been addressed by various project; however, it is not yet adequately solved, due to its complexity. In URBANITE, the constructed model will be used to predict and classify traffic flow changes based on the provided changes in the new policies. Policy-makers will select the defined KPI's that need to be evaluated by the validation engine and based on the scores the new policies achieve, policy-makers will be able to make an informed decision about which policies should be deployed in the city.

2.4 Advanced Visualization Methods

Another important task will be the implementation of advanced visualizations for mobility patterns, highlighting important events, and results of policy validations. The main visualization functionalities will present the information on a combination of map layers, describing where in the city specific events or a sequence of events occurred. Visualizations will involve the use of heat maps, traffic flow graphics, and other transportation clusters. Users will be able to change and interact with the visualization parameters. For example, select specific time ranges, zoom, highlight, display additional information, etc. Considering the variety and characteristics of the data, one concern is regarding the depicting multidimensional data in a human-perceivable manner. Several graphical methods are customarily used for a preliminary analysis of generic multivariate datasets [2]: scatter plots, pie charts and bar plots, histograms, box plots, violin and bean plots, spider/radar/star/polar plots, glyph plots, mosaic and spine plots, treemaps, and others.

Traffic datasets are generally high-dimensional or spatial-temporal [3], thus visualizing traffic data mostly employs information visualization and visual analytics.

Traffic data contain multiple variables, of which the most important ones are time and space. Several different types of visualisation are currently used for traffic data, among them: visualization of time, visualization of spatial properties and spatio-temporal visualization.

Location is the main spatial property of traffic data. Based on the aggregation level of location information, visualization of spatial properties can be categorized into three classes: point-based visualization (no aggregation), line-based visualization (first-order aggregation), and region-based visualization (second-order aggregation) [3].

Heatmaps are the most used visualisation tools to show the integrated quantity of a large scale of objects in a map.

A preliminary user interface prototype is depicted in Figure 2.

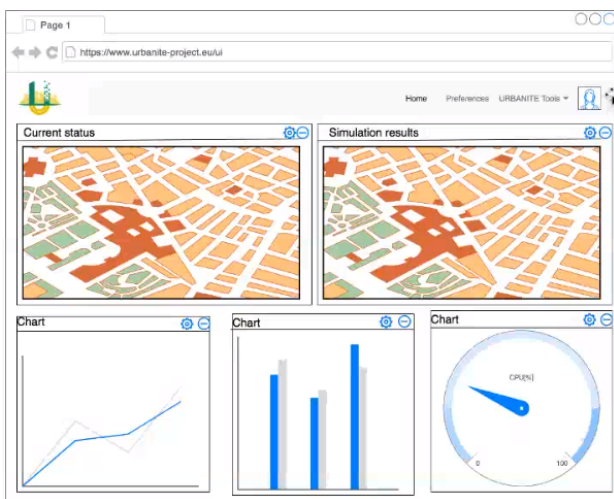


Figure 2: User Interface Mock up of the URBANITE Platform.

3 DATA SOURCES

There are several collection procedures of the traffic related data and they range from sensor readings to airborne imagery and

social media data [13]. The involvement of the municipalities of Bilbao, Helsinki, Amsterdam and Messina will provide a wide range of data sources related to the urban mobility, along with the public, open-source ones.

Several types of data sources were identified for the URBANITE project:

- geospatial data, e.g. maps (Open Street Maps ¹, but also proprietary maps of the cities)
- additional info such as: car and lorry registration, information on parking lots, dynamic parking data, cadastre information, commercial register, care services, tourism accommodation
- demographics: statistical information on the number of inhabitants of different city districts, the number of households, population's age brackets, city boundaries, etc.
- public transportation: tram and metro lines, static and dynamic information about the public bus transport service, the GPS position of the buses
- traffic data: the count of car traffic and speeds, traffic status in real time, vehicle counts on the ring roads, etc.
- bicycle information: bike counters, bicycle collection points, calculated number of bikes in specific road segments, City-Bikes ²
- pedestrian: manual counts of pedestrians
- electric charging stations
- taxi stops available
- harbour transport data, ferry traffic statistics
- geographic airport information
- air quality (OpenAQ ³)
- noise maps
- wheather data (OpenWeatherMap ⁴)

The format of this datasets varies from JSON, XML, CSV, XLSX, WMS, GEOJSON or GML. The main issue with the mobility related data sources it is related to the high level of heterogeneity, both in terms of data format and data availability. Most of the cities involved on the project have some data related to the traffic in the city, for example, but the format of the data, the level of granularity (how often is the data updated) and the availability of historical data (for how long does the city store historical data) varies greatly from one case to another.

Another special aspect that needs to be addressed is the impact of the COVID-19 on the mobility sector. Since COVID-19 has disrupted all of the social, economic and political aspects of life, the urban mobility area was also affected. Some analysis [1] revealed that the overall mobility fall was up to 76%, public transport users dropped by up to 93%, NO₂ emissions were reduced by up to 60%, and traffic accidents were reduced by up to 67% in relative terms. This phenomenon of experiencing unexpected change of concepts or data characteristics over time is referred to as concept drift [7] and is one of the key challenges that the URBANITE project will need to deal with when choosing the best way to proceed for making the most appropriate predictions regarding the impact of various traffic policies. The algorithms developed should take into consideration the stability-plasticity dilemma as a reference. Especially since it's still difficult to predict how the crisis derived from the pandemic will evolve and how the urban mobility will be afterwards.

¹<https://www.openstreetmap.org/>

²<https://api.citybik.es/v2/>

³<https://openaq.org/>

⁴<https://openweathermap.org/>

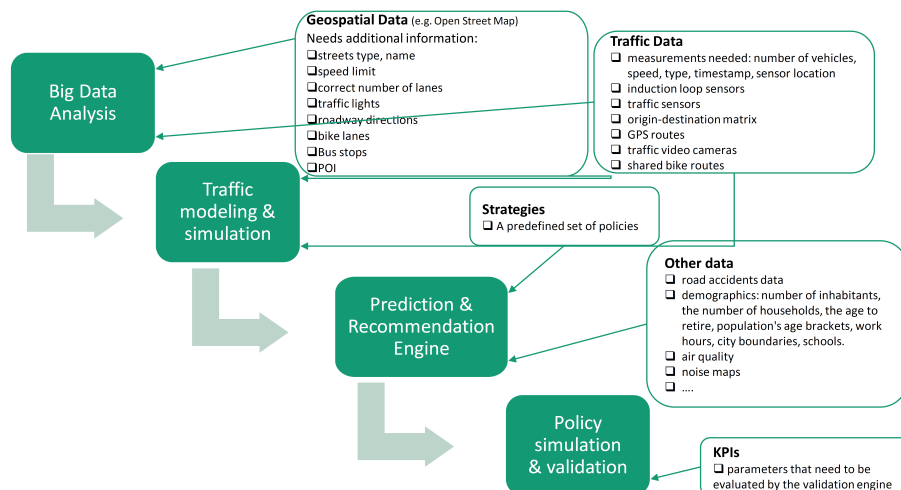


Figure 3: Data Sources for the URBANITE Platform.

4 CONCLUSIONS

The technical core in the URBANITE project focuses on the development of advanced AI algorithms for analysis of big data on mobility. The developed methods and tools will provide substantial support for policy-makers to tackle complex policy problems on the mobility domain and will enable their validation on case-specific models. The goal of the activities is to implement novel tools and services in order to enable policy-makers to use advanced data analysis and machine learning methods during the design of novel policies for a specific city.

One underlining factor in URBANITE is the adaptation of everything that it is created to civil servants, citizens and interesting parties that may or not be digitally literate. The use of big data techniques and artificial intelligence algorithms, up till now, is not a common skill among public servants and this is one of the reasons the data analysis processes and user interaction mechanisms described in this work are developed with the abilities of the non-experts in mind too.

ACKNOWLEDGMENTS

This paper is supported by European Union's Horizon 2020 Research and Innovation Programme, URBANITE project under Grant Agreement No.870338.

REFERENCES

- [1] Alfredo Aloí, Borja Alonso, Juan Benavente, Rubén Cordera, Eneko Echániz, Felipe González, Claudio Ladisa, Raquel Lezama-Romanelli, Álvaro López-Parra, Vittorio Mazzei, et al. 2020. Effects of the covid-19 lockdown on urban mobility: empirical evidence from the city of santander (spain). *Sustainability*, 12, 9, 3870.
- [2] Sunith Bandaru, Amos HC Ng, and Kalyanmoy Deb. 2017. Data mining methods for knowledge discovery in multi-objective optimization: part a-survey. *Expert Systems with Applications*, 70, 139–159.
- [3] Wei Chen, Fangzhou Guo, and Fei-Yue Wang. 2015. A survey of traffic data visualization. *IEEE Transactions on Intelligent Transportation Systems*, 16, 6, 2970–2984.
- [4] Ricardo Colomo-Palacios, Francisco José García-Peñalvo, Vladimir Stantchev, and Sanjay Misra. 2017. Towards a social and context-aware mobile recommendation system for tourism. *Pervasive and Mobile Computing*, 38, 505–515.
- [5] F.O. Isinkaye, Y.O. Folajimi, and B.A. Ojokoh. 2015. Recommendation systems: principles, methods and evaluation. *Egyptian Informatics Journal*, 16, 3, 261–273. ISSN: 1110-8665. DOI: <https://doi.org/10.1016/j.eij.2015.06.005>.
- [6] Joseph A Konstan and John Riedl. 2012. Recommender systems: from algorithms to user experience. *User modeling and user-adapted interaction*, 22, 1-2, 101–123.
- [7] Jesus L Lobo, Javier Del Ser, Miren Nekane Bilbao, Ibai Lana, and Sancho Salcedo-Sanz. 2016. A probabilistic sample matchmaking strategy for imbalanced data streams with concept drift. In *International Symposium on Intelligent and Distributed Computing*. Springer, 237–246.
- [8] Chenguang Pan and Wenxin Li. 2010. Research paper recommendation with topic analysis. In *2010 International Conference On Computer Design and Applications*. Volume 4. IEEE, V4–264.
- [9] Nymphia Pereira and Satishkumar L Varma. 2019. Financial planning recommendation system using content-based collaborative and demographic filtering. In *Smart Innovations in Communication and Computational Sciences*. Springer, 141–151.
- [10] Miquel Angel Piera, Roman Buil, and Egils Ginters. 2013. Validation of agent-based urban policy models by means of state space analysis. In *2013 8th EUROSIM Congress on Modelling and Simulation*. IEEE, 403–408.
- [11] Tomasz Rutkowski, Jakub Romanowski, Piotr Woldan, Paweł Staszewski, Radosław Nielek, and Leszek Rutkowski. 2018. A content-based recommendation system using neuro-fuzzy approach. In *2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, 1–8.
- [12] Diego Sánchez-Moreno, Ana B Gil González, M Dolores Muñoz Vicente, Vivian F López Batista, and María N Moreno García. 2016. A collaborative filtering method for music recommendation using playing coefficients for artists and users. *Expert Systems with Applications*, 66, 234–244.
- [13] G. Zhou, J. Lu, C.-Y. Wan, M. D. Yarvis, and J. A. Stankovic. 2008. *Body Sensor Networks*. MIT Press, Cambridge, MA.

Towards End-to-end Text to Speech Synthesis in Macedonian Language

Marija Neceva, Emilija Stoilkovska, Hristijan Gjoreski
mneceva@gmail.com, emi.stoilkovska@gmail.com, hristijang@feit.ukim.edu.mk
Faculty of Electrical Engineering and Information Technologies
Ss. Cyril and Methodius University
Skopje, N. Macedonia

ABSTRACT

A text-to-speech (TTS) synthesis system typically consists of multiple stages: text analysis frontend, an acoustic model and an audio synthesis module. Building these components often requires extensive domain expertise and may contain brittle design choices. The paper presents an end-to-end deep learning approach to speech synthesis in Macedonian language. The developed model uses the Google's Tacotron architecture and is able to generate speech out of text from multiple speakers using attention mechanism. It consists of three parts: an encoder, an attention-based decoder and a post-processing network. The model was trained on a dataset recorded by five, mixed gender speakers, resulting in 25.5 hours of data, or 13,101 pairs of text-speech segments. The results show that the model successfully generates speech from text data, which was empirically shown using a quantitative questionnaire answered by 42 subjects.

KEYWORDS

text-to-speech, deep learning, tacotron, multi-speaker, seq2seq, text, audio, attention

1 INTRODUCTION

Modern TTS pipelines are complex [1]. For example, statistical parametric ones have a text frontend, extracting various linguistic features, a duration model, an acoustic feature prediction model and a complex signal-processing-based vocoder [2][3]. These components usually require extensive domain expertise, are laborious to design and must be trained independently. Consequently, errors from each component may compound. Otherwise, implementing an integrated end-to-end TTS system offers many advantages. First, it can be trained on <text, audio> pairs with minimal human annotation. It also alleviates the need for laborious feature engineering. Further, it allows rich conditioning on various attributes, such as speaker or language, or high-level features like sentiment. Similarly, adaptation to new data might also be easier. Finally, a single model is likely to be more robust than a multi-stage. All these advantages imply that an end-to-end system allows training on huge amounts real world data. But knowing that TTS is a large-scale inverse problem and due to existence of different pronunciations or speaking styles, decompressing a highly compressed source text into audio may cause difficulties in the learning task of an end-to-end model. The main problem is coping with large variations at the signal level for a given input. Moreover,

unlike end-to-end speech recognition [4] or machine translation [5], TTS outputs are continuous, and much longer than input sequences. Mainly referring to the advantages of end-to-end systems, this paper proposes an implementation of Google's Tacotron model as a TTS system for Macedonian language. Tacotron is an end-to-end generative TTS model based on the sequence-to-sequence model (seq2seq) [6] with attention paradigm [7]. This model takes characters as input and outputs raw spectrogram. We implemented our own version of Tacotron, based on few published articles. What we kept is their deep learning architecture, but made some changes in model's hyper parameters and other utilities (like known symbols, numbers etc.). That way the model was adapted to work with Cyrillic. Given <text, audio> pairs, our Tacotron model was trained completely from scratch only on our dataset. It does not require phoneme-level alignment, so it can easily scale to using large amounts of acoustic data with transcripts.

2 RELATED WORK

WaveNet [8] is a powerful, non end-to-end, generative audio model which works well for TTS synthesis. It is used as a replacement of the vocoder and acoustic model of the system. It can be slow due to its sample-level autoregressive nature. It also requires conditioning on linguistic features from an existing TTS frontend.

Deep Voice [9] is a neural model which replaces every component in a typical TTS pipeline by a corresponding neural network. However, each component is independently trained, and it's nontrivial to change the system to train in an end-to-end fashion.

Wang et. al [10] presents one of the first studies of end-to-end TTS using seq2seq with attention. However, it requires a pre-trained hidden Markov model (HMM) aligner to help the seq2seq model learn the alignment and a vocoder due to predicting vocoder parameters. Furthermore, the model is trained on phoneme inputs with possibilities of hurting the prosody and producing limited experimental results.

Char2Wav [11] is an independently developed end-to-end model that can be trained on characters. However, it still predicts vocoder parameters before using a SampleRNN neural vocoder [12] and their seq2seq and SampleRNN models need to be separately pre-trained.

MAIKA [26] is a Macedonian TTS project that was made public few months ago. However, there is no documentation of how it works. Therefore, it is technically challenging to

compare with a system that only has web interface which generates sound.

eSpeak [27] is an open source TTS project that also supports Macedonian language. The documentation states that the Macedonian model is based on the Croatian - which has its limitations since the Macedonian language is quite different, especially the pronunciation and the grammar.

3 MODEL ARCHITECTURE

The backbone of Tacotron is a seq2seq model with attention [7][13]. Figure 1 illustrates the model, which includes an encoder, an attention-based decoder, and a post-processing net. At a high-level, this model takes characters as input and produces spectrogram frames, which are later converted to waveforms. These components are described below.

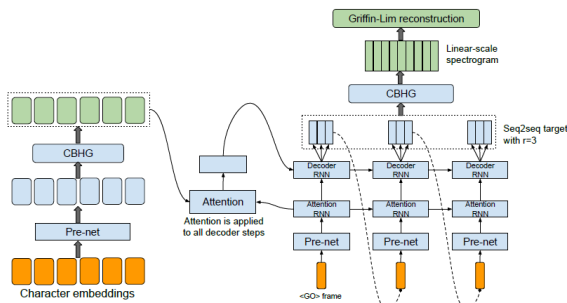


Figure 1: Model architecture

3.1 CBHG Module

CBHG is a module for extracting representations from sequences. It consists of bank of 1-D convolutional filters, followed by highway networks [14] and a bidirectional gated recurrent unit (GRU) [15]. The input sequence is first convolved with k sets of 1-D convolutional filters. These filters explicitly model local and contextual information (creating unigrams, bigrams, up to k -grams). Next the convolution outputs are stacked together and max pooled along time to increase local invariances. Further the processed sequence is passed to a few fixed-width 1-D convolutions, whose outputs are added with the original input sequence via residual connections [16]. Batch normalization [17] is used for all convolutional layers. Moreover, the fixed-width convolution outputs are fed into a multi-layer highway network to extract high-level features. Finally, a bidirectional GRU RNN has been stacked on top, extracting sequential features from both forward and backward context.

3.2 Encoder

The encoder extracts robust sequential representations of text. The input to the encoder is a character sequence, with each character represented as a one-hot vector and embedded into a continuous vector. Onto each embedding is applied a set of non-linear transformations, known as “pre-net”. The “pre-net” is represented as a bottleneck layer with dropout, helping convergence and improving generalization. A CBHG module transforms the “pre-net” outputs into the final encoder representation used by the attention module. Moreover, CBHG-based encoder reduces overfitting and makes fewer mispronunciations than a standard multi-layer RNN encoder.

3.3 Decoder

Tacotron model uses a content-based \tanh attention decoder [18], where a stateful recurrent layer produces the attention query at each decoder time step. The input of decoder’s RNN is formed by concatenating the context vector and the attention RNN cell output. Decoder’s internal structure is a stack of GRUs with vertical residual connections [5], used for speeding up convergence. A simple fully-connected output layer is used to predict the decoder targets. Its target is 80-band mel-scale spectrogram, later converted to waveform by a post-processing network. It predicts multiple, non-overlapping, output frames at each decoder step. Predicting r frames at once divides the total number of decoder steps by r , which reduces model size, training and inference time and increases convergence speed. This is likely because neighboring speech frames are correlated and each character usually corresponds to multiple frames, plus emitting multiple frames allows the attention to move forward early in training. For defining the input of the next decoding step “teacher forcing” mechanism is used, pointing that on each time step, decoder’s input is the ground-truth value of the previous predicted decoder output.

3.4 Attention Mechanism

Attention mechanism is applied in order to “learn” mappings between input and output sequences through gradient descent and back-propagation. It is used as a way for the decoder to learn at which time step, which internal state of the encoder deserves more attention when generating its current output. The whole process of calculating the attention weights and using them to form the decoder input has been illustrated in Figure 2.

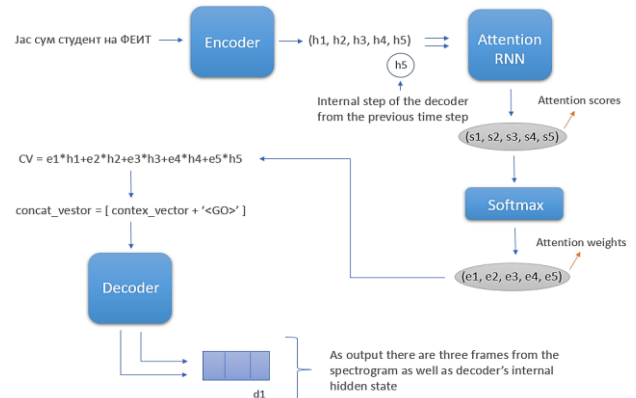


Figure 2: What is behind the attention mechanism

3.5 Post-processing Net and Waveform Synthesis

The post-processing net is converting the seq2seq target to a form that can be synthesized into waveforms [20][21]. Since Griffin-Lim has been used as a synthesizer, the post-processing net learns to predict spectral magnitude, sampled on a linear-frequency scale. The Griffin - Lim algorithm allows convergence towards estimated phase layer. Phase’s quality depends on the number of iterations applied. Although more iterations may lead to overfitting, better audio is produced. Within our setup, Griffin-Lim converges after 50 iterations even though 30 iterations seems to be enough.

3.6 Model Parameters

The log magnitude spectrogram is obtained using Hann windowing with 50 ms frame length, 12.5 ms frame shift, and 2048-point FT. 24 kHz sampling rate has been used for all experiments. For both seq2seq decoder (mel-scale spectrogram) and post-processing net (linear-scale spectrogram) a simple L1 loss with equal weight has been used. The model has been trained using a batch size of 4, where all sequences are padded to a max length.

4 DATASET

There is no public dataset of audio data in Macedonian language, therefore we had to create one. We used publicly available books in Macedonian from the website of the National Association of the Blind of the Republic of North Macedonia. The books have been recorded by 5 speakers, 3 male and 2 female. They are segmented using an algorithm which separates input audio based on silence length and threshold. Silence length varies between 700 – 1000 ms. The audio clips were additionally padded with 700 ms at both beginning and end to avoid sudden cut offs.

Next, the audio files were transcribed manually, aided by the written version of the audio book. The transcriptions are void of any punctuation, capitalization, or any special characters, including numbers. They include only the 31 letters from the Macedonian alphabet and the space character to separate between words. The reason for this is that the initial dataset was also used for another task (Speech Recognition) and the researchers removed the punctuations. In this phase we could not retrieve the original raw data that includes the punctuation. The final dataset contains 13,101 audio files and transcripts in Macedonian language [25]. Additional statistics about the dataset are listed in Table 1.

To be mentioned, the goal of the dataset is not the dataset itself, but how we can develop a deep learning, end to end, multi-speaker TTS for Macedonian language. Detailed language analysis of the dataset is planned for another study, in which the focus will be more on the linguistically part of the dataset.

Table 1: Dataset statistics

Total Clips	13 101
Total Words	188 521
Distinct Words	28 791
Total duration	25:36:20
Mean Clip Duration	7.04 sec
Min Clip Duration	0.73 sec
Max Clip Duration	97.6 sec (1.37 min)

5 TRAINING AND EVALUATION

5.1 Training

During the training phase there is an output produced on every 1000th step. It takes few seconds for an output to be produced. Each output contains five files, three of which give

information about the model formed up to that step, while the other two are an alignment plot and an audio file synthesized by that mode. The synthesized audio file is used for checking the quality of the current model. The alignment plot shows if the decoder has learned which input state of the encoder is important for producing its current output. That means if there is an “A” on input, “A” should be produced as sound for output. As a good alignment plot is considered the one who looks like a diagonal line. This system was trained for 5 days, reached 412 000 steps and got 412 different models. The system started showing a good alignment on 63 000th step. The last model was chosen as referent one. Its training and test results sound much better and were more understandable than those generated from the other models.

5.2 Evaluation

To estimate the model’s performance, we used 10, out of 14 random sentences as test examples. The results show that more than half of the synthesized audio files [22] were successfully representing the input sequence of the model. This was empirically shown using a quantitative questionnaire [23] answered by 42 subjects, 10 IT experts and 32 general public volunteers. The questionnaire was made up of 10 stages, for each of the 10 audio files. The reason for choosing 10 test examples was to make the questionnaire more compact, smaller and quicker for the evaluators. Each stage contains 3 sub questions for the currently observed audio file. The Mean Opinion Score (MOS) [24] was used as a measure for answering i.e. scoring each one of it. MOS is a measure of audio quality. It is a subjective measurement used to test the listener’s perception of the audio quality and clarity. A group of 42 subjects were asked to do the questionnaire. Each audio file required to be scored with a score from 1-5 in terms of three criterions: naturalness, intelligibility and accuracy. Where naturalness stands for the similarity of produced audio file with the natural human speech, intelligibility or clarity of spoken words and accuracy or how much the spoken sequence corresponds with the original, required to be spoken text.

The results from the questionnaire are shown in Table 2. Each row of the table represents the MOS for one of the three criterions, calculated separately for experts and volunteers. The calculations are done by summing the scores for each criterion and consequently averaging it. By analyzing the results for each criterion is clear that, the experts score the model’s performance better compared to the volunteers. Looking at the total score, experts evaluated the model’s performance for 0.265 better than the volunteers. We speculate that the reason for this might be that when the experts are evaluating the model they also take into account the technical challenges and aspects of such system. On the other hand the volunteers simply evaluate the sound and its quality.

Additionally, in Figure 3 and Figure 4 we show the box-plots for the answers given by the experts and the volunteers respectively. The figures show that the accuracy is the characteristic that achieves the highest score, and the naturalness is the characteristic that achieved the lowest score. We speculate that the reason for low naturalness score is the presence of sudden pauses when words should be spoken or existence of mumbling instead of clear pronunciation. There are only few such occurrences.

Table 2: MOS Score results

	MOS Score	
	Experts	Volunteers
Accuracy	4.8	4.6
Intelligibility	4.5	4.2
Naturalness	4.1	3.9
Total	4.5	4.2

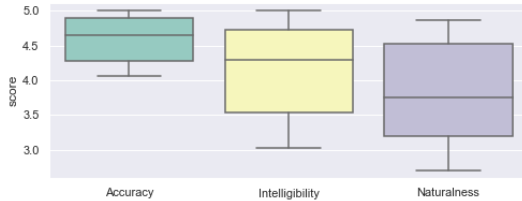


Figure 3: Box plot of all grades given by the volunteers



Figure 4: Box plot of all grades given by the IT experts

6 CONCLUSION

The paper presented an end-to-end deep learning approach to speech synthesis in Macedonian language. The developed model uses the Google’s Tacotron architecture and generates speech out of text from multiple speakers using attention mechanism. The approach consists of three parts: an encoder, an attention-based decoder and a post-processing network. The model was trained on a dataset recorded by five, mixed gender speakers, resulting in nearly 25.5 hours of data. The results show that the model successfully generates speech from text, which was empirically shown using a quantitative questionnaire answered by 42 subjects.

To the best of our knowledge, this is the first end-to-end multi-speaker deep learning model for Macedonian language. We strongly believe that this will be a benchmark and motivation for future studies and finally to have a decent TTS system for Macedonian - which has significant societal impact.

Some of the limitations of the model are the gender diversity of speakers and the limited dataset. There is definitely room for improvement, and probably the dataset plays a crucial role in it. However, the data collection process is extensive and very time consuming task. With the given dataset we cannot estimate or empirically evaluate how much more data is needed to achieve state-of-the-art intelligibility and naturalness of artificially created speech. Additionally, in a few of the generated samples there are pauses at places where a word should be spoken. The reason for this is when the model generates sound, it uses character embeddings with specific ordering, learned during training. If those embeddings have never been seen during training, the model

will not be able to properly pronounce them. Note that this is not the case with all of the words not being present in the training data, but in very rare occasions. Normally, the model will still generate speech even though a word is not present in the dataset.

ACKNOWLEDGEMENT

We are thankful for the support of the NVIDIA Corporation and their generous donation of a Titan XP GPU.

REFERENCES

- [1] P.Taylor. Text-to-speech synthesis. Cambridge university press, 2009.
- [2] H. Zen, K.Tokuda,A.W.Black. Statistical parametric speech synthesis. *Speech Communication*, 51(11):1039–1064, 2009.
- [3] Y.Agiomyrgiannakis. Vocode the vocoder and applications in speech synthesis. In *Acoustics, Speech and Signal Processing (ICASSP)*, 2015 IEEE International Conference on, pp. 4230–4234. IEEE, 2015.
- [4] W.Chan, N.Jaitly, Q.Le, and O.Vinyals. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP)*, 2016 IEEE International Conference on, pp. 4960– 4964. IEEE, 2016.
- [5] Y.Wu, M.Schuster, Z.Chen, Q.V.Le, M.Norouzi,W.Macherey, M.Krikun, Y.Gao, Q.Gao, K.Macherey. Google’s neural machine translation system: Bridging the gap between human and machine translation. arXiv:1609.08144, 2016.
- [6] I.Sutskever, O.Vinyals,Q.V.Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pp. 3104–3112, 2014.
- [7] D.Bahdanau, K.Cho, Y.Bengio. Neural machine translation by jointly learning to align and translate. arXiv:1409.0473, 2014.
- [8] A.Oord, S.Dieleman, H.Zen, K.Simonyan, O.Vinyals, A.Graves, N.Kalchbrenner, A.Senior, K.Kavukcuoglu. WaveNet: A generative model for raw audio. arXiv:1609.03499, 2016.
- [9] S.Arik, M.Chrzanowski, A.Coates, G.Diamos, A.Gibiansky, Y.Kang, X.Li, J.Miller, J.Raiman, S.M.Shoeibi. Deep voice: Realtime neural text-to-speech. arXiv:1702.07825, 2017.
- [10] W.Wang, S.Xu, B.Xu. First step towards end-to-end parametric TTS synthesis: Generating spectral parameters with neural attention. In *Proceedings Interspeech*, pp. 2243–2247, 2016.
- [11] J.Sotelo, S.Mehri, K.Kumar, J.F.Santos, K.Kastner, A.Courville, Y.Bengio. Char2Wav: End-to-end speech synthesis. In *ICLR2017 workshop submission*, 2017.
- [12] S.Mehri, K.Kumar, I.Gulrajani, R.Kumar, S.Jain, J.Sotelo, A.Courville, Y.Bengio. SampleRNN: An unconditional end-to-end neural audio generation model. arXiv preprint:1612.07837, 2016.
- [13] O.Vinyals, Ł.Kaiser, T.Koo, S.Petrov, I.Sutskever, G.Hinton. Grammar as a foreign language. In *Advances in Neural Information Processing Systems*, pp. 2773–2781, 2015.
- [14] R.K.Srivastava, K.Greff, J. Schmidhuber. Highway networks. (2015).
- [15] J.Chung, C.Gulcehre, K.H.Cho, Y.Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv:1412.3555, 2014.
- [16] K.He, X.Zhang, S.Ren, J.Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.770–778, 2016.
- [17] S.Ioffe, C.Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167, 2015.
- [18] O.Vinyals, Ł.Kaiser, T.Koo, S.Petrov, I.Sutskever, G.Hinton. Grammar as a foreign language. In *Advances in Neural Information Processing Systems*, pp. 2773–2781, 2015.
- [19] D.Kingma, J.Ba. Adam: A method for stochastic optimization. *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2015.
- [20] Y.Masuyama, K.Yatabe, Y.Koizumi, Y.Oikawa, N.Harda (2019): Deep Griffin – Lim Iteration.
- [21] J.Wodecki (2018): Intuitive explanation of the Griffin – Lim algorithm.
- [22] Synthesized test audio files: https://drive.google.com/drive/folders/1LkgKAKcD9qNMw_3stbHEhszxhrPyPmAA?usp=sharing.
- [23] Quantitative questionnaire used for evaluation of the model: https://docs.google.com/forms/d/e/1FAIpQLSeJJVRjU3tzblLi1mix9buN-Os002GFaTvSp9TVO7520CPNUvA/viewform?fbclid=IwAR1bLE8hrEALj7MwHkAgDKrF0jfyCID-DTuCiGdJ8Nc68Jl1XYv_1_MRxoE.
- [24] P.C. Loizou. *Speech Quality Assessment*. University of Texas-Dallas, Department of Electrical Engineering, Richardson, TX, USA.
- [25] M.Trajanoska, H.Gjoreski. Towards end-to-end Speech Recognition in Macedonian Language. *BalkanCom* (2019).
- [26] MAIKA: <https://maika.mk/>
- [27] eSpeak: <http://espeak.sourceforge.net/>

Improving Mammogram Classification by Generating Artificial Images

Ana Peterka[†]
University of Ljubljana,
Faculty of Computer and
Information Science,
Ljubljana, Slovenia
anapeterka1151@gmail.com

Zoran Bosnić
University of Ljubljana,
Faculty of Computer and
Information Science,
Ljubljana, Slovenia
zoran.bosnic@fri.uni-lj.si

Evgeny Osipov
Luleå University of Technology,
Department of Computer Science,
Electrical and Space Engineering,
Luleå, Sweden
evgeny.osipov@ltu.se

ABSTRACT

Training a deep convolutional neural network (DCNN) from the scratch is difficult, because it requires large amounts of labeled training data. This is a big problem especially in the medical domain, since datasets are scarce and data is often imbalanced. This can result in overfitting the model. Fine-tuning a model that has been pre-trained on a large dataset shows promising results. Another approach is to augment the dataset with artificially generated learning examples. In this paper, we augment the learning set with artificially generated images that are produced by conditional infilling GAN. The results that we obtained show that we can relatively easily generate realistically looking mammograms that improve the classification of benign and malignant mammograms.

KEYWORDS

data augmentation, transfer learning, CNN, ResNet-50, GAN, ciGAN

1 INTRODUCTION

Breast cancer is a cancer that is found in the tissue of the breast, when abnormal cells grow in an uncontrolled way. It can affect both women and men, though it is prevalent in women. Statistics show that it has the highest mortality rate of any cancer in women worldwide and that 1 in 8 women in the EU will develop breast cancer before the age of 85¹. Screening mammography helps diagnose cancer at an early stage, which significantly increases the survival rates. However, the evaluation of mammograms performed by doctors and radiologists is tedious, lengthy and error prone, as it results in a high number of false positives.

New approaches in deep learning (DL), in particular convolutional neural networks (CNNs), have proven their potential for medical imaging classification tasks. This could relieve radiologists and give patients quicker and more accurate diagnosis. However, the performance of CNNs are dependent on large labeled datasets, which are hard to obtain in the medical

imaging field due privacy concerns of the patients and the time consuming expert annotations. Furthermore, the data is often imbalanced, meaning that pathologic findings are relatively very rare. This can result in overfitting the model and bad generalization ability.

So far, this problem has been addressed with transfer learning and data augmentation techniques. In this paper, we evaluate these techniques on the CBIS-DDSM dataset, which is a publicly available dataset that contains benign and malignant mammograms. We propose a novel approach of generating new images with Generative Adversarial Networks (GANs) combined with traditional data augmentation, such as horizontal flipping, rotations etc., and evaluate if increasing the dataset helped to achieve better classification. We also test if fine tuning a ResNet-50 model helps improve the results.

The paper is structured as follows. Section 2 presents the related work, Section 3 describes the data augmentation techniques used, Section 4 the training process, Section 5 the evaluation metrics used and the results, and in Section 6 we state our conclusions and discuss the prospective future work.

2 RELATED WORK

This section provides a brief review of past work that falls down to three categories:

1. improved classification with traditional data augmentation,
2. improved classification with generating synthetic images using generative adversarial network,
3. transfer learning and fine tuning.

The problem with small datasets, especially in the medical domain, is that models that are trained on them tend to overfit the data. There are a lot of approaches to reduce it, like batch normalization, dropout, data augmentation and also transfer learning. Traditional data augmentation based on affine transformations, such as translation, rotation, shearing, flipping and scaling, is the most widely used and very easy to implement. They are ubiquitous in computer vision tasks and show very promising results [1]. However, they do not bring any new visual features that could additionally improve the generalization of the CNN.

Synthetic image generation with GANs enables more variability to the dataset and further improves robustness of the

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
Information Society 2020, 5–9 October 2020, Ljubljana, Slovenia
© 2020 Copyright held by the owner/author(s).

¹ <https://www.europadonna.org/breast-cancer-facs/>

classification network. GANs were inspired by game theory, where two neural networks are pitted against each other using a minmax strategy. They were first introduced in [2], and they have recently been applied to many different medical imaging applications, mostly for image to image translation and image inpainting. In [3], the authors used conditional infilling GAN to synthesize lesions on mammograms.

Transfer learning and fine tuning for mammography medical images was the main topic in [4] and [5]. In [4], they demonstrated that a whole image model trained on DDSM can be easily transferred to INbreast without using its lesion annotations and using only a small amount of training data. In [5], the authors showed that fine tuning ResNet-50 model pre-trained on ImageNet can be used to perform tumor classification in CBIS-DDSM dataset.

In this paper, we will first use traditional data augmentation techniques and later additionally augment the dataset with applying the ciGAN (conditional infilling GAN). We will evaluate the improvements with a fine tuned ResNet-50 model.

3 AUGMENTING THE DATASET

In this section, we first describe the dataset, then we explain the traditional data augmentation methods used and a GAN method for synthesizing new images.

3.1 The CBIS-DDSM dataset

CBIS-DDSM [6] is a publicly available dataset that contains digitized images from scanned films of mammogram images and it is a subset of the DDSM dataset that consists of only benign and malign cases, while the DDSM also contains normal. The data was acquired from 1566 patients and it contains both mediolateral oblique (MLO) and craniocaudal (CC) views of each breast. Images are grayscale, and they have corresponding binary masks that indicate mass and ROI images of that mass.

Images are in DICOM format, which is the standard for medical imaging information. The data is already split in the training and testing set. We used a part of the testing set as a validation set for the classification network.

3.2 Traditional data augmentation

To compensate for the lack of training images, we used classical data augmentation techniques, in particular horizontal flipping, rotations of up to 30°, and zoom range from 0.75 to 1.25 and test if this improved the performance of the CNN.

3.3 Data augmentation with GANs

To further augment and balance the dataset, we use a GAN variant, called conditional infilling GAN (ciGAN) [3]. GANs are a type of generative models, which means they are able to produce novel examples, based on the training data. They consist of two neural networks, a generator and a discriminator, which are pitted against each other. Generator tries to capture the data's distribution while the discriminator tries to distinguish real and generated examples. By training them simultaneously, the generator will get better at generating realistic data, while the discriminator gets better at distinguishing real and fake data. In the case of ciGAN, the generator is based on a cascaded refinement network (CRN) [8], where features are generated at multiple scales before being concatenated, which yields a more realistic image synthesis.

In our approach, we apply the ciGAN to sample a location on a healthy mammogram and then synthesize a lesion in its location, as shown in Figure 1. The input is a concatenated stack of:

- a corrupted image (one channel grayscale image with lesion replaced by uniform distribution of values between 0 and 1),
- a binary mask that marks lesion (1 representing the location of the lesion, and the zeros elsewhere), and

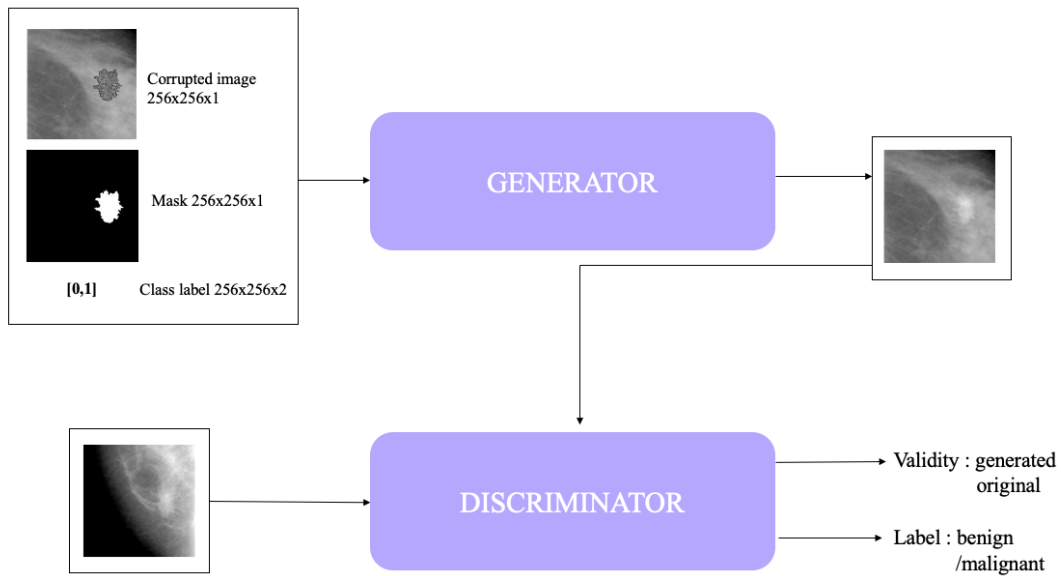


Figure 1: The ciGAN architecture. The input consists of two one channel images, and 2 class channels for indicating malignant/benign label. Output of the generator is, together with the real image fed into the discriminator, which predicts whether each image is either generated or original and also whether the image contains benign or malignant lesions.

- the class label ([1,0] representing the non-malignant class, and [0,1] representing the malignant class).

The generator is comprised of multiple convolutional blocks. The first convolutional block receives input stack, downsampled to the 4x4 resolution. Resolution is doubled between consecutive blocks. So the next convolutional block is fed with concatenation of the output from the first layer, upsampled to the 8x8 and an input stack resized to 8x8. This is repeated until resolution of 256x256 is obtained. The discriminator has similar, but inverse structure.

3.4 Differences to the related work

Our work is based on the before mentioned ciGAN [3], with a few improvements. While the former method was trained on non-malignant versus malignant cases, our approach uses benign and malignant cases, since we believe that the real hardship is distinguishing the lesions and not only noticing them. Images in the original work show that for acquiring synthetic non-malignant mammograms, the lesion was removed, making the picture a normal mammogram. Since we used a sliding window approach of extracting normal patches instead of the mask, we did not have to remove the malignant lesion, but we applied both masks independently, so we obtained only benign and malignant cases. All generated benign cases contain a lesion. We also applied zooming and rotation to lesions before generating new images, hence our generated images have more diverse tumors.

4 GENERATING ARTIFICIAL IMAGES

4.1 Preprocessing

To extract patches of 256x256 pixels that are fed into ciGAN, we used a sliding window technique. The program loops through the whole mammogram image with the stride of 128 and checks if the rectangular region overlaps the majority of the breast. It also checks whether the patch contains lesion or it shows only normal breast tissue, and labels it accordingly. This is done by comparing the same region of the corresponding binary mask. At the end the patch dataset contains 5466 images, 1743 of them are normal, 2198 benign and 1525 malignant.

After acquiring a dataset of patches, the program loops through all the patches containing only normal tissue. For each normal patch, it randomly chooses one patch that contains a lesion. The patch with lesion is then randomly zoomed in/out by a small factor, to obtain more diverse masses. Next, we check whether on the same location as is lesion, on the normal patch, is only breast tissue and not background. If not, the next random lesion patch is chosen and the whole process is repeated until a suitable match is found.

Once there is a suitable pair obtained, the normal image is corrupted, by replacing the area defined by the mask of the lesion with uniform distribution.

4.2 Loss functions

The ciGAN model is trained by utilizing three loss functions [3]:

- Perceptual loss: is a loss calculated between the ground truth and the output image. But unlike a per-pixel loss, which is based on differences between pixels, it measures the discrepancy between high-level perceptual features

extracted from pretrained networks [10]. It encourages the generator to output images with similar high level features as the original image. In this case, the VGG-19 [11] convolutional neural network is used, pretrained on the ImageNet dataset. It is defined as

$$L(R, S) = \sum_l \|\phi(L)_l - \phi(S)_l\|_1$$

where R denotes a real image, S a synthetic image and ϕ a feature function;

- Boundary Loss: is used to encourage smoothing between infilled components and the context of the generated image. It is a L1 difference between the real and generated images at the boundary and defined as

$$B(R, S) = \|w \odot (R - S)\|_1$$

where w denotes the mask with Gaussian filter of standard deviation 10 applied, and \odot is the element wise product;

- Adversarial Loss: is the general GAN loss. It is defined as a distance between the true and the generated distribution at the current iteration. Its goal is to converge to the equilibrium in the minmax game between generator G and discriminator D, as follows:

$$L(G, D) = E_{c,R}[\log D(c, R)] + E_R[\log(1 - D(c, S))]$$

where c denotes the class label.

4.3 Training

The ciGAN is first pretrained on perceptual loss for 300 epochs. Then the training of discriminator and generator are alternating, when loss for either drops below 0.3 for additional 2000 epochs. The ciGAN produces realistic images as shown in Figure 2.

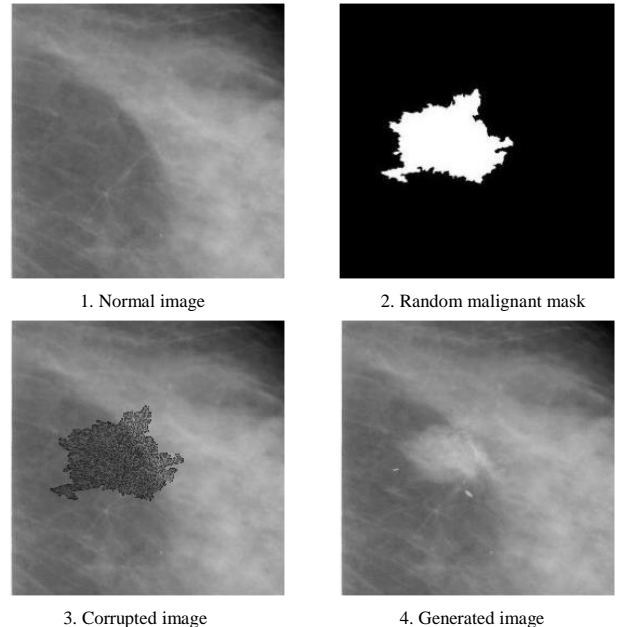


Figure 2: A generated sample from ciGAN. Image 1 is the normal image without a lesion, image 2 is the binary mask representing the random malignant lesion, image 3 is the corrupted image and image 4 is the synthesized image with malignant lesion.

5 EVALUATION AND RESULTS

For evaluation of results three metrics were used. The first one is accuracy, which tells us how many examples were correctly classified. The second one is recall/sensitivity, which is the fraction between true positives and the sum of true positives and false positives. It is the most important metric in this case, due to the risk of overlooking cancer. The third one is Area Under Curve (AUC), which measures area under the ROC curve. We evaluate the results by performing 4 experiments:

1. **Shallow CNN** [12]: we implement it as the baseline. The network is fed a patch and classifies it as either malignant or benign. It consists of three convolutional blocks, composed of 3x3 Convolutions, Batch Normalization, ReLU activation function and Max Pooling, followed by three Dense layers, and softmax function for binary classification.
2. **ResNet-50**: we classify the data using a ResNet-50 [13].
3. **ResNet-50** with finetuning: we check if transfer learning improves the results.
4. **ResNet-50 + Traditional** data augmentation,
5. **ResNet-50 + Traditional** data augmentation and generated **artificial** images.

As mentioned in [5], we fine-tuned the Resnet-50 [12] model with ImageNet weights. It is an extremely deep neural network with 150+ layers and consists of convolutional layers, pooling layers and multiple residual blocks. In the residual blocks, the layers are fed into the next layer and also directly into the layers about two to three hops away. The input to the ResNet-50 model is a patch of a size 224x224x3. Since mammograms have only grayscale channels, the color information is copied over all three channels. We used the Adam optimizer with an initial learning rate of 10^{-5} , $\beta_1 = 0.9$, $\beta_2 = 0.999$, $e = 10^{-8}$ and ImageNet weight initialization. We trained it for 50 epochs with batch size of 32 and a 0.9 learning rate decay every 30 epochs.

Table 1 shows the obtained results. We can see that already using only fine tuning using ResNet-50 improved the results. After combining ResNet-50 with traditional data augmentation, we obtained even better performance metrics. Nevertheless, by increasing the dataset with relatively small amounts of synthetic images while simultaneously balancing it, we improved accuracy and AUC even more, but obtaining a slight decrease in the recall.

6 CONCLUSION

In this paper we discussed overcoming the obstacle of small and imbalanced mammography dataset. We proposed an approach for artificial generation of images that are produced by a conditional infilling GAN (ciGAN). The results showed that we can relatively easy generate realistically looking mammograms that improve the classification of benign and malignant mammograms. Further, we evaluated the learning performance when using fine-tuning, classical data augmentation and synthetic examples. The results showed that each of these techniques improved classification, yielding the best results using all three together.

Comparing the results to previously developed method [3], we obtained worse results in terms of AUC, but we believe the reason behind it is the fact that all our images contain lesion, which must be harder for a neural network to distinguish, compared to distinguishing non-malignant and malignant images.

Testing these methods on different medical datasets shall be the subject of future work. As well, one may consider using these methods on bigger data sets and improve the current state of the art algorithms. Since the ciGAN's discriminator was also conditioned on class, we intend on extracting its features and using it for classification on other mammography dataset, for example on the INBreast dataset. We also plan on adding more synthetic images to the dataset, to see if we can further improve the classification.

Currently, the mammogram classification is performed by the doctors and radiologists, but we hope that improving the classification with the use of machine learning combined with these and similar techniques could relieve them of such tasks in the near future.

Table 1: The obtained accuracy, recall and AUC scores

	accuracy	recall	AUC
Shallow CNN	0.57267	0.44810	0.54943
Resnet-50 without finetuning	0.58295	0.53859	0.58634
ResNet-50	0.60155	0.55769	0.59443
ResNet-50 + traditional	0.67132	0.64231	0.66666
ResNet-50 + traditional + artificial	0.76145	0.61538	0.71638

REFERENCES

- [1] Wang, J., & Perez, L. (2017). The effectiveness of data augmentation in image classification using deep learning. *Convolutional Neural Networks Vis. Recognit*, 11.
- [2] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems* (pp. 2672-2680).
- [3] Wu, E., Wu, K., Cox, D., & Lotter, W. (2018). Conditional infilling GANs for data augmentation in mammogram classification. In *Image Analysis for Moving Organ, Breast, and Thoracic Images* (pp. 98-106). Springer, Cham.
- [4] Shen, L. (2017). End-to-end training for whole image breast cancer diagnosis using an all convolutional design. *arXiv preprint arXiv:1711.05775*.
- [5] Agarwal, R., Diaz, O., Lladó, X., & Martí, R. (2018, July). Mass detection in mammograms using pre-trained deep learning models. In *14th International Workshop on Breast Imaging (IWBI 2018)* (Vol. 10718, p. 107181F). International Society for Optics and Photonics.
- [6] Lee, R. S., Gimenez, F., Hoogi, A., Miyake, K. K., Gorovoy, M., & Rubin, D. L. (2017). A curated mammography data set for use in computer-aided detection and diagnosis research. *Scientific data*, 4, 170177, <https://wiki.cancerimagingarchive.net/display/Public/CBIS-DDSM>.
- [7] Odena, A., Olah, C., & Shlens, J. (2017, July). Conditional image synthesis with auxiliary classifier gans. In *International conference on machine learning* (pp. 2642-2651).
- [8] Chen, Q., & Koltun, V. (2017). Photographic image synthesis with cascaded refinement networks. In *Proceedings of the IEEE international conference on computer vision* (pp. 1511-1520).
- [9] Johnson, J., Alahi, A., & Fei-Fei, L. (2016, October). Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision* (pp. 694-711). Springer, Cham.
- [10] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- [11] Lévy, D., & Jain, A. (2016). Breast mass classification from mammograms using deep convolutional neural networks. *arXiv preprint arXiv:1612.00542*.
- [12] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

Mobile Nutrition Monitoring System: Qualitative and Quantitative Monitoring

Nina Reščič
nina.rescic@ijs.si
Department of Intelligent Systems,
Jožef Stefan Institute
International Postgraduate School
Jozef Stefan
Ljubljana, Slovenia

Marko Jordan
Department of Intelligent Systems,
Jožef Stefan Institute
Ljubljana, Slovenia

Jasmijn de Boer
ConnectedCare
Nijmegen, Netherlands

Ilse Bierhoff
ConnectedCare
Nijmegen, Netherlands

Mitja Luštrek
mitja.lustrek@ijs.si
Department of Intelligent Systems,
Jožef Stefan Institute
Ljubljana, Slovenia

ABSTRACT

The WellCo project¹ aims to provide a mobile application featuring a virtual coach for behaviour changes aiming to achieve for healthier lifestyle. The nutrition monitoring module consists of two main parts - qualitative (Food Frequency Questionnaire) and quantitative (eating detection and bite counting). In this paper we present the nutrition monitoring module that connects both monitoring aspects as implemented in the virtual coach (mobile application).

KEYWORDS

nutrition monitoring, eating detection, FFQ

1 INTRODUCTION

Proper nutrition habits are beneficial for healthy lifestyle and help to prevent many chronic diseases, such as cancer, diabetes and hypertension. Automated monitoring has become really important in nutrition monitoring, but in only gives quantitative information (when is the user eating, how much did he eat...), while qualitative information (what is the user eating) is acquired by using 24 hour food recall diaries or by using Food Frequency Questionnaires (FFQs). In the WellCo project we aimed to develop a user friendly nutrition module, which monitors qualitative and quantitative aspects of users' nutrition. We combined the self-reported FFQ, Extended Short Form Food Frequency Questionnaire (ESFFQ), developed and validated in the project project [5], with automated monitoring by using a commercially available wearable smartwatch. This paper describes the developed module and the improvements we made since our previous papers [5, 2, 7].

By using wrist-worn devices to collect data, it is possible to recognize eating gestures [4] or even count 'bites' or assess caloric intake [10]. Mirtchou et al. [3] explored eating detection by using several sensors and combining real-life and laboratory data.

¹<http://wellco-project.eu>

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

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

© 2020 Copyright held by the owner/author(s).

Edison et al. [8] proposed a method that recognizes each intake gesture separately and later the intake gestures within 60 minutes interval are clustered.

For qualitative monitoring we evaluated both dietary recalls and FFQs as self-reporting methods. However, dietary recalls require typing or complex food item selection which can be cumbersome on mobile devices, so we opted for FFQ. FFQs are the most commonly selected tools in nutrition monitoring as they are efficient, cost-effective and non-invasive [9, 6]. The developed FFQ covers all key aspects of healthy diet, and is modular, so that only questions pertaining to certain aspects can be asked. This is important in ubiquitous settings where one wishes to minimize the required inputs from the user.

To our knowledge the developed application module is the first one to combine qualitative (validated FFQ) and quantitative monitoring (bite counting method) and to provide recommendations based on data gathered by monitoring.

2 METHOD

2.1 Method Overview

The paper describes the nutrition monitoring module developed in the Wellco project.

The **qualitative monitoring** starts with a five-question questionnaire that provides essential information about the user's diet. Based on this, some goals to improve the user's nutrition can already be recommended. However, the users are invited to answer a more extensive questionnaire that paints a more complete picture and allows recommending more goals. This questionnaire is an extended version of a validated questionnaire, and the extension was validated by us [5]. How successful the users are at achieving their goals is monitored with goal-specific questions on a bi-weekly basis.

The **quantitative monitoring** uses the accelerometer and gyroscope in a smartwatch to detect micromovements related to eating (e.g., picking up food, putting it into the mouth). From a sequence of such micromovement, we then recognise whether the user has made one "bite" (taken the food to the mouth). The improved method uses a Convolutional neural network to recognise the micromovements and a LSTM neural network to recognise bites. The latter achieved higher accuracy so it was the one selected to be integrated into the WellCo system.

2.2 FFQ - Qualitative Monitoring

When choosing goals that would help users of the WellCo virtual coach towards behavioural changes for healthier lifestyle, we were leaning on national dietary recommendation and dietary recommendations for elderly, combined with expert knowledge by the nutritionist involved in the project. A summary of national dietary recommendations is presented in Table 1.

Guidelines specifically for the elderly are very similar to national dietary recommendations for all three countries involved in pilots (Italy, Spain and Denmark), but they put additional emphasis on dairy consumption, as this is a good source of proteins and calcium, which are beneficial and often under-consumed; drinking enough water, as dehydration is often a problem with elderly; and leucine consumption (in milk, peanuts, oatmeal, peanuts, fish, poultry, egg white, wheat sprouts, etc). Given these recommendations, we chose goals we will suggest WellCo users to follow and use in order to improve their diet: *fruit consumption, vegetable consumption, salt consumption, fat consumption, fibre consumption, protein consumption, salt consumption, fish consumption and water consumption*.

In our search for a comprehensive but still short FFQ we found a validated questionnaire named Short Food Frequency Questionnaire (SFFQ)[1], which consists of 23 questions and fully covers five of our chosen goals – *fruit and vegetable consumption, sugar consumption, fat consumption and fish consumption*. To cover the four missing goals (protein, fibre, salt and water consumption) we added additional 8 questions, turning the SFFQ into the so-called Extended Short Food Frequency Questionnaire (ESFFQ). The validation of the questionnaire is described in our previous paper [5].

2.3 Quantitative Monitoring

The main objective of the smartwatch-based nutrition monitoring is bite counting (counting the number of time the user takes food to the mouth).

The bite-counting algorithm described in [2] was used as the base for all of the following work. When deciding how to present the results of the developed algorithm to the users in the mobile application, we had to make some improvements to our model. As the number of bites does not really give much useful information to the users, we decided to join individual bites into meals and to recognize meals as *snack, small meal or big meal*.

2.3.1 Datasets. To construct the bite detection algorithm, we created the Wild Meals Dataset (WMD). It includes 51 sessions and 99 meals, with known starting and ending time points, belonging to 11 unique subjects, recorded 'in-wild'. For 68 of those meals we have also obtained the approximate number of the corresponding bites, since the subjects were asked to count them while eating. Additionally we used the publicly available The Food Intake Cycle (FIC) dataset and The Free Food Intake Cycle (FreeFIC). All datasets contains tri-axial signals from accelerometers and gyroscopes in wrist devices with the sampling frequency of 100 Hz.

2.3.2 Meal detection method. The algorithm for meal detection was comprised of two parts: in the first part probabilities that given time periods are part of eating were assigned, whereas in the second part these probabilities were grouped together to form a meal.

First we linearly interpolated all accelerometer and gyroscope measurements as well as the probabilities of bites to 4Hz frequency. Next, the normalization was applied to interpolated accelerometer and gyroscope data. We constructed 90 s long sliding windows with a 2.5s step. Each window contained 360 of the previously obtained accelerometer, gyroscope and bite probability values (obtained with CNN and LSTM networks as described in [2]). 4Hz frequency was used to achieve faster training and predicting, while also enabling us to construct longer windows. A window was labelled as a positive instance, if the majority of the window belonged inside a meal.

To solve this machine learning task, an inception-type neural network was constructed, with the added GRU layers at the end. The inception part of the network is mainly made of two types of inception blocks. Both types consist of convolutional layers and end with a filter concatenation. The B block includes also a max pooling operation. Each block in the network is succeeded by a max pooling layer. The entire architecture is presented in Table 1. The inputs were transformed in the (batch size, timestamps, 1, 7) shape. "Prep" (preparation) in Table 1 refers to the yellow convolutional layers in Figure 5, whereas "Pool proj" refers to 1x1 convolutional layer after 4x1 max pooling layer. The final model used approximately 130 K parameters.

With the intention of smoother and better learning, the ratio between positive and negative instances was fixed to 1:2. During the sampling, we actually focused more on problematic areas, by first predicting with the network and then selecting problematic instances to train on. Learning rate was set to keep decreasing every few epochs. Certain hyper-parameters were subject to optimization during cross-validation, with the help of hyperopt library. The function to minimize was categorical cross entropy.

In the next part, the outputs $\in [0,1]$ of the neural network, which represent the probabilities that the given windows are eating instances, are taken to form possible/candidate meals. This is done in the following manner:

- **Round 1:** Find all probabilities, denoted as beacons, that are higher than a p_1 threshold. Include also all probabilities that are closer than t_1 seconds to any of the beacons. Set all the other probabilities temporarily to 0.
- **Round 2:** Find all probabilities that are higher than a p_2 threshold and group them together, if they are immediately next to each other. For each group find the time distance to its nearest group. Finally remove all groups that have either 1 or 2 members and are more than t_2 seconds away from the corresponding nearest group.
- **Round 3:** If there exist any two groups of the form $[A,B]$ and $[C,D]$, where $0 \leq C - B \leq t_3$ (all in seconds), combine these two groups together to form a new group, $[A,D]$. This means that indices in $[A,D]$ can now represent the probabilities of zero as well.
- **Round 4:** Similar as Round 3, but with a t_4 parameter in place of t_3 .

At this point the probabilities of windows, previously temporarily set to zero, are switched back to their original values. For the final model, we obtained the following values of the above hyperparameters:

Since $p_2 > p_1$, this means that Round 1 in this particular case was not necessary, although in some other cases it could have been. Once the candidate meals have been obtained, the features are constructed for the ensemble of random forest, support vector machine, knn and gradient boosting algorithms. The ensemble

Table 1: Architecture of the network

Type	Units/Nodes	Kernel/stride	Output	1x1	4x1 prep	4x1	6x1 prep	6x1	Pool
Inception-A			360x1x128	32		64		32	
Max pool		3x1/2	180x1x128						
Inception-B			180x1x128	32	64	64	16	16	16
Max pool		3x1/2	90x1x128						
Inception-B			90x1x128	32	64	64	16	16	16
Max pool		3x1/2	45x1x128						
Inception-B			45x1x128	32	64	64	16	16	16
Max pool		3x1/2	23x1x128						
GRU			23x32						
GRU			32						
Dense	64		64						
Dropout(0.36)			64						
Dense	2		2						

Table 2: Hyperparameters.

p1	t1(sec)	p2	t2(sec)	t3(sec)	t4(sec)
0.46	61	0.87	120	63	61

makes the final decision whether a candidate meal is in fact a meal or not. The following features are created for each candidate meal:

- The mean, standard deviation, the 25th, 50th and 75th percentile of all the probabilities inside a given candidate meal.
- The mean and standard deviation of the first and second half of a potential meal, separately.
- The mass of all the future probabilities inside all the potential meals closer than 3 hours to a given candidate meal, divided by their time centre.
- The mass of all the past probabilities inside all the potential meals closer than 3 hours to a given candidate meal, divided by their time centre.

Hyper-parameters for each model in the ensemble, as well as p1, t1, p2 t2, t3 and t4 values, were calculated with a cross-validation, with the help of hyperopt library. The function to minimize was negative F1-score.

3 RESULTS

3.1 Bite Counting

In Table 4 we present the results of evaluation of our work. The analysis of the entire pipeline is based on Leave-One-Subject-Out double cross-validation. For calculation of the above statistics the following definitions were used:

- True positive prediction of a meal: any prediction of the respective meal for which the majority of the prediction laid inside the ground truth meal. If there was more than one prediction of eating for a certain meal, only one prediction is actually counted as a true positive, whereas all the others are not regarded as a false positive.. This is due to the possibility that the subjects didn't eat their entire recording time; as such it did not seem reasonable to penalize the pipeline for predicting more than one meal, however, only one true positive is counted in order not to encourage the algorithm to predict a bundle of eating instances.

Table 3: Results of bite recognition and meal detection algorithm.

	F1-score	precision	recall	cov_area	outside_area
Avg.	0.76	0.88	0.72	0.81	0.03

Table 4: Example of recommendations for qualitative monitoring (*goal_sugar*) and quantitative monitoring (*nutrition_number_of_meal*).

goal_sugar	It seems you don't eat enough vegetables. Vegetables are important sources of many nutrients, such as vitamins, minerals and dietary fibre. Try to eat 2 servings of vegetables per day. Serving is 1 cup of fresh or half cup of cooked vegetables.
nutrition_number_of_meal	Try to eat 3–5 meals per day (e.g. 3 bigger, 2 smaller). Avoid snacking between meals.

- For F1-score, precision and recall, def A was used, while cov_area and outside_area used def B. However, double cross-validation results show that all ground truth meals, with one exception, had at most one corresponding, true positive predicted meal.
- Covered area (cov_area): for a given ground truth meal, the length of the areas, which laid inside the ground truth meal, of the corresponding true positive meals, divided by the length of the ground truth meal.
- Outside area (outside_area): for a given predicted, true positive meal, the length of the area that laid outside the corresponding ground truth meal, divided by the length of the predicted meal.

3.2 Application Implementation

The application shows users the detected meals, number of bites and score quality for the chosen goals (see Figure 1). Based on the results we additionally show the user recommendations to follow in order to improve their nutrition. Example for recommendations for both, qualitative and quantitative monitoring is shown in table.

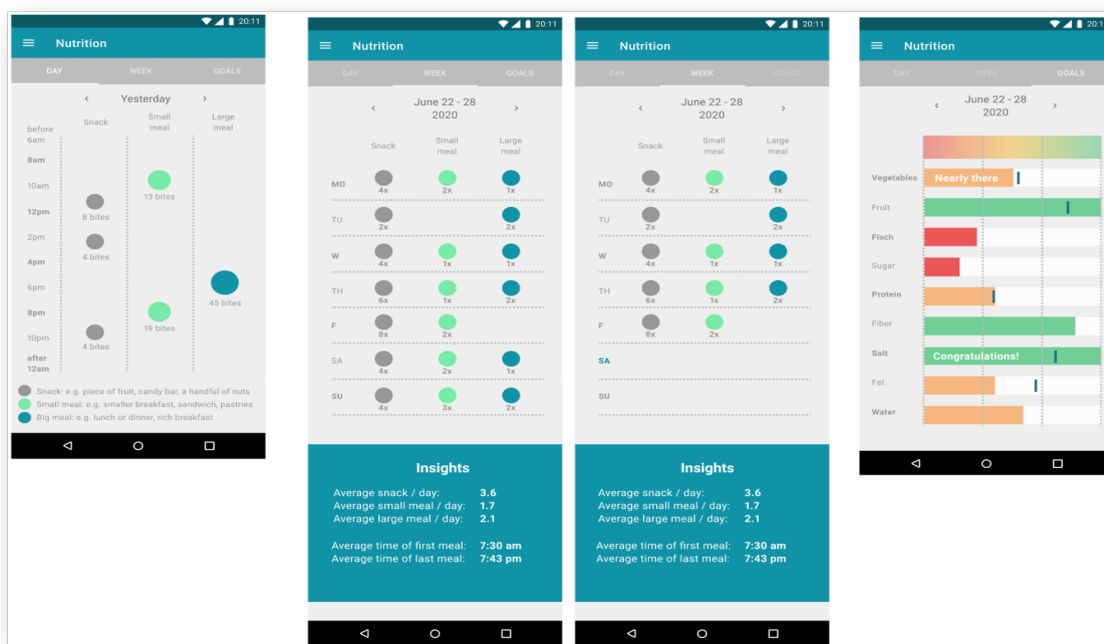


Figure 1: Application view for both monitoring tasks.

4 CONCLUSION

The developed nutrition monitoring module consists of two parts - qualitative monitoring and quantitative monitoring. Both of the developed modules are implemented in a mobile application. In our future work we would like to improve the developed eating detection and bite counting algorithms.

The developed FFQ (ESFFFQ) can be used to support a wide range of nutrition goals and minimizes the number of questions asked, so it is suitable for mobile nutrition monitoring. To make the application user friendly the questions from the FFQ will not be asked all at the same time, but separately during a course of fortnight. This means that some of the questions won't be asked, hence it is really important to ask the right questions. In our future work we will try to explore the problem of question ranking. With this we would be able to ask the questions in a specific order and lose as few information as possible.

5 ACKNOWLEDGMENTS

WellCo Project has received funding from the European Union's Horizon2020 research and innovation program under grant agreement No 769765.

REFERENCES

- [1] Christine L Cleghorn, Roger A Harrison, Joan K Ransley, Shan Wilkinson, James Thomas, and Janet E Cade. 2016. Can a dietary quality score derived from a short-form ffq assess dietary quality in uk adult population surveys? *Public Health Nutrition*, 19, 16, 2915–2923. DOI: 10.1017/S1368980016001099.
- [2] 2019. Counting bites with a smart watch. In *Slovenian Conference on Artificial Intelligence : proceedings of the 22nd International Multiconference Information Society*. Volume A, 49–52.
- [3] Mark Mirtchouk, Drew Lustig, Alexandra Smith, Ivan Ching, Min Zheng, and Samantha Kleinberg. 2017. Recognizing eating from body-worn sensors: combining free-living and laboratory data. 1, 3. DOI: 10.1145/3131894.
- [4] Raul I. Ramos-Garcia, Eric R. Muth, John N. Gowdy, and Adam W. Hoover. 2014. Improving the recognition of eating gestures using intergesture sequential dependencies. *IEEE Journal of Biomedical and Health Informatics*, 19, 3, 825–831.
- [5] Nina Reščič, Eva Valenčič, Enej Mlinarič, Barbara Koroušič Seljak, and Mitja Luštrek. 2019. Mobile nutrition monitoring for well-being. In *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers (UbiComp/ISWC '19 Adjunct)*. London, United Kingdom, 1194–1197.
- [6] JS Shim, K Oh, and HC Kim. 2014. Dietary assessment methods in epidemiologic studies. *Epidemiol Health*, 36. DOI: 10.4178/epih/e2014009.
- [7] Simon Stankoski, Nina Reščič, Grega Mežič, and Mitja Luštrek. 2020. Real-time eating detection using a smartwatch. In Junction Publishing, USA.
- [8] Edison Thomaz, Irfan Essa, and Gregory D. Abowd. 2015. A practical approach for recognizing eating moments with wrist-mounted inertial sensing. In *Association for Computing Machinery*, New York, NY, USA. ISBN: 9781450335744. DOI: 10.1145/2750858.2807545.
- [9] Frances Thompson and T Byers. 1994. Dietary assessment resource manual. *The Journal of nutrition*, 124, (December 1994), 2245S–2317S. DOI: 10.1093/jn/124.suppl_11.2245s.
- [10] Shibo Zhang, William Stogin, and Nabil Alshurafa. 2018. I sense overeating. *Inf. Fusion*, 41, C, (May 2018), 37–47. DOI: 10.1016/j.inffus.2017.08.003.

Recognition of Human Activities and Falls by Analyzing the Number of Accelerometers and their Body Location

Miljana Shulajkovska, Hristijan Gjoreski
miljanash@gmail.com, hristijang@feit.ukim.edu.mk
Faculty of Electrical Engineering and Information Technologies
Ss. Cyril and Methodius University
Skopje, N. Macedonia

ABSTRACT

This paper presents an approach to activity recognition and fall detection using wearable accelerometers placed on different locations of the human body. We studied how the location and the number of wearable accelerometers influence on the performance of the recognition of the activities and the falls. The final goal was to build a machine learning model that can correctly recognize the activities and the falls using as few accelerometers as possible. The model was evaluated on a public dataset consisting of more than 850 GB of data, recorded by 17 people. In total we evaluated 15 combinations of four accelerometers placed on the belt, the left ankle, the left wrist and the neck. The results showed that the neck and the ankle accelerometers proved sufficient to correctly recognize all the activities and falls with 94.2% accuracy. Each of the sensors used individually achieved 94.02% and 93.4% accuracy respectively.

KEYWORDS

activity recognition, fall detection, wearable sensors, machine learning

1 INTRODUCTION

According to United Nations World Population Prospects 2019, by 2050, one in six people in the world will be over the age of 65 [1]. As people are getting older, their risk for falls also increases. Falls are a major public health problem in elderly people often causing fatal injuries. It is important to assure that injured people receive assistance as quickly as possible. Because of this, building a good fall detection system is of a big importance to help medicine solve this problem.

The field of Human Activity Recognition (HAR) and fall detection has become one of the trendiest research topics due to availability of low cost, low power consuming sensors, i.e., accelerometers. The recognition of human activities has been approached in two different ways, namely using ambient and wearable sensors [2]. In the former, the sensors are fixed in predetermined points of interest on the body of the subject, so the inference of activities entirely depends on

* Both authors contributed equally to this research.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

Information Society 2020, 5-9 October 2020, Ljubljana, Slovenia
© 2020 Copyright held by the owner/author(s).

the voluntary interaction of the users with the sensors. In the latter, the sensors are attached to the user.

This paper presents a machine learning approach to activity recognition and fall detection using wearable accelerometers placed on different locations of the human body. The goal of the paper is to study how the location and the number of wearable accelerometers influence on the performance of the recognition of the activities and the falls. This study is of practical importance of such systems, i.e., to build a machine learning model that can correctly recognize the activities and the falls using as few accelerometers as possible.

2 RELATED WORK

A considerable amount of work has been done in human activity recognition for the last decade where a lot of studies aim to identify activities based on data obtained from accelerometers as sensors widely integrated into wearable systems [3][4].

Researchers have reported high accuracy scores in detecting activities when investigating the best placement of the accelerometer on the human body [5][6][7]. Increasing the number of sensors increases the complexity of the classification problem. For these reasons, a number of studies have investigated the use of a single accelerometer. However, doing so generally decreases the number of activities that can be recognized accurately [8]. Consequently, one of the major considerations in activity recognition is the location or combination of locations of the accelerometers that provide the most relevant information.

In [5] the authors study the best location to place accelerometers for fall detection, based on the classification of postures. Four accelerometers were placed at the chest, waist, ankle and thigh. Statistical features were calculated for each axis of the accelerometer in addition to the magnitude. Results indicated that one accelerometer (chest or waist) by itself was not enough to sufficiently classify the activities (75%). There was, however, a significant improvement in classification accuracy achieved by combining the accelerometer at the chest or waist with one placed on the ankle (91%). Following the work described in [5] we explore this approach using different dataset while investigating all possible sensor placement combinations.

3 ACTIVITY RECOGNITION

3.1 Dataset

In this research we used the UP-Fall Detection dataset, which is publicly available [9]. The dataset contains 17 Subjects that are performing 11 activities. Each activity is performed 3 times. The activities performed are related to six simple human daily activities and five human falls showed in Table 1. These types of activities and falls are chosen from the analysis of those reported in literature [10][11]. All daily activities are performed during 60 s, except jumping that is performed during 30 s and picking up an object which it is an action done once within a 10-s period. A single fall is performed in each of the three ten seconds period trials.

Table 1: Activities performed in the Dataset

Activity ID	Description	Duration (s)
1	Falling forward using hands	10
2	Falling forward using knees	10
3	Falling backwards	10
4	Falling sideward	10
5	Falling sitting in empty chair	10
6	Walking	60
7	Standing	60
8	Sitting	60
9	Picking up an object	10
10	Jumping	30
11	Laying	60

In order to collect data from young healthy subjects without any impairment, is considered a multimodal approach for sensing the activities in three different ways using wearables, context-aware sensors and cameras, all at the same time. However, of our particular interest is how acceleration data can be used for the recognition of activities. The analyzed data is obtained from accelerometers placed on ankle, neck, wrist and belt. This way we created 15 different

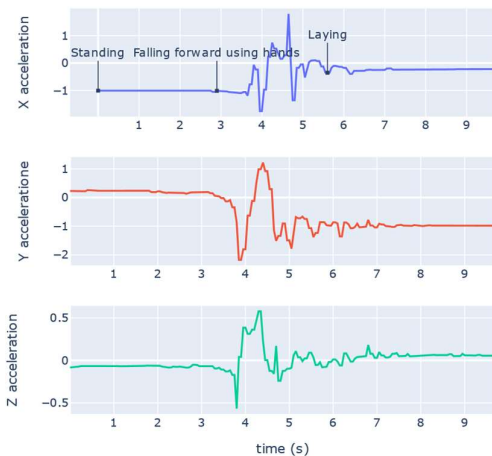


Figure 1 Raw Data from 3-Axis Accelerometer

datasets representing every combination of these sensors to show the importance of the placement of the accelerometer.

In our research the sampling rate of the sensor is 18 Hz, which means 18 samples are provided every second. In Figure 1 **Error! Reference source not found.** the raw data from 3-axis accelerometer is shown from person who is performing three activities: standing, falling forward using hands and laying.

3.2 Feature Extraction

Feature extraction is really important step in the activity recognition process in order to filter relevant information and obtain quantitative measures that allow signals to be compared. In our research we used statistical features to create the feature vectors. All the attributes are computed by using the technique of overlapping sliding windows [5].

Because the final sampling frequency of our accelerometers was 18 Hz, we chose a window size of 18, which is one second time interval. We decided for one-second time interval because in our target activities there are transitional activities (standing up and going down) that usually last from one to four seconds. Statistical attributes are extracted for each axis of the accelerometer.

The feature extraction phase produces 36 features (summarized in Table 2) from the accelerations along the x, y, and z axes. The first three features (Mean X/Y/Z,) provide information about body posture, and the remaining features represent motion shape, motion variation, and motion similarity (correlation).

Once the features are extracted (and selected), a feature vector is formed. During training, feature vectors extracted from training data are used by a machine learning algorithm to build an activity recognition model. During classification, feature vectors extracted from test data are fed into the model, which recognizes the active.

Table 2: Overview of the extracted features. The number of features is represented with #

Feature name	#
Mean (X, Y, Z)	3
Standard deviation (X, Y, Z)	3
Root mean square (X, Y, Z)	3
Maximal amplitude (X, Y, Z)	3
Minimal amplitude (X, Y, Z)	3
Median (X, Y, Z)	3
Number of zero-crossing (X, Y, Z)	3
Skewness (X, Y, Z)	3
Kurtosis (X, Y, Z)	3
First Quartile (X, Y, Z)	3
Third Quartile (X, Y, Z)	3
Autocorrelation (X, Y, Z)	3

3.3 Methods

Machine learning approach was used for the activity recognition. In this study, the machine learning task is to learn a model that will be able to classify the target activities

(e.g. standing, sitting, falling, etc.) of the person wearing accelerometers. For this purpose, we used 4 different machine learning algorithms: Random Forest, Support Vector Machine, k-Nearest Neighbors and Multilayer Perceptron.

The Random Forest (RF) classifier, like its name implies, consists of a large number of individual decision trees that operate as an ensemble. The fundamental concept behind RF is the low correlation between any of the individual constituent models protecting each other from their individual error.

The Support Vector Machine (SVM) method has also been broadly used in HAR although they do not provide a set of rules understandable to humans. SVMs rely on kernel functions that project all instances to a higher dimensional space with the aim of finding a linear decision boundary (i.e., a hyperplane) to partition the data.

The k-Nearest Neighbors (k-NN) is a supervised classification technique that uses the Euclidean distance to classify a new observation based on the similarity (distance) between the training set and the new sample to be classified.

The Multilayer Perceptron (MLP) [12], is an artificial neural network with multilayer feed-forward architecture. The MLP minimizes the error function between the estimated and the desired network outputs, which represent the class labels in the classification context. Several studies show that MLP is efficient in non-linear classification problems, including human activity recognition. Brief study of MLP and other classification methods is shown in [13][14].

4 EXPERIMENTS

4.1 Evaluation Techniques

To properly evaluate the models, we divided the data into train and test using leave-one-person-out cross-validation. With the leave-one-person-out each fold is represented by the data of one person. This means the model was trained on the data recorded for 16 people and tested on the remaining person's data. This procedure was repeated for each person data (17 times) and the average performance was measured.

Four evaluation metrics are commonly used in activity recognition: the recall, precision, accuracy and F-measure. We have analyzed the accuracy score, which shows how many of the predicted activities are correctly classified.

4.2 Results

For the first experiment we compared 4 ML models using the ankle accelerometer - shown in Figure 2. We used the ankle accelerometer because our initial studies showed that it performs the best. Random Forest showed the best results with 92.92% of accuracy. Therefore, it was used for further experiments.

Table 3 shows the comparison of activity recognition accuracy using 4 accelerometers placed on ankle, belt, neck and wrist. It shows how the number and placements of accelerometer can affect the recognition of particular activities.

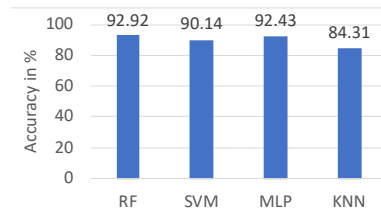


Figure 2: Comparison of different algorithms using Ankle Accelerometer

Placing the accelerometer on the belt can distinguish sitting, standing or jumping, but distinguishing different kind of falls that include some transitions, like standing, falling and then laying is a problem. Adding one accelerometer on the neck, can slightly improve the results, but still cannot recognize correctly the falls. Combination of neck and ankle accelerometer proved best results with 94.2% accuracy. On the other hand, an accelerometer on the ankle can distinguish walking, standing and laying, but has problems with picking up an object and also recognizing the falls. Most of the fall activities are recognized as standing or laying. By combining

Table 3: Comparison of activity recognition accuracy using different number of accelerometers (1, 2, 3 or 4) placed on ankle, belt, neck and wrist

Activities	1				2		3		4		5		6		7	
	Ankle	Belt	Neck	Wrist	Ankle+ Belt	Ankle+ Neck	Ankle+ Wrist	Belt+ Neck	Belt+ Wrist	Neck+ Wrist	Ankle+ Belt+ Neck	Ankle+ Belt+ Wrist	Ankle+ Neck+ Wrist	Belt+N eck+ Wrist	Ankle+ Belt+ Neck+ Wrist	
Falling forward using hands	57.9	64.4	75.1	63.0	63.2	73.6	60.3	71.7	63.2	70.1	69.5	63.1	72.7	71.1	73.5	
Falling forward using knees	72.6	81.4	76.2	55.7	76.4	77.7	61.2	77.6	68.6	64.5	77.9	75.8	75.0	72.4	78.7	
Falling backwards	69.4	68.7	71.4	52.4	71.7	75.0	64.4	70.6	56.6	62.2	71.7	67.0	70.5	63.9	69.1	
Falling sideways	63.6	67.3	69.7	42.5	66.8	74.8	58.1	67.3	53.4	57.4	70.0	68.3	70.3	62.1	70.4	
Falling sitting in empty chair	56.8	68.3	75.6	48.7	65.8	71.4	52.8	70.6	58.3	67.3	73.5	60.4	69.6	70.0	71.7	
Walking	98.6	96.6	99.2	94.2	98.9	98.6	98.9	96.6	98.5	98.7	98.8	99.0	98.6	98.6	98.8	
Standing	96.6	91.4	69.6	92.8	97.7	97.1	91.9	93.1	93.5	96.8	98.3	98.2	97.8	97.9	98.2	
Sitting	90.2	66.4	95.0	85.2	75.9	84.5	80.6	83.9	68.4	87.6	71.4	72.3	79.9	85.5	76.8	
Picking an object	67.7	73.0	86.8	43.3	76.0	88.0	63.0	82.1	71.3	82.9	87.7	74.1	87.0	82.1	86.8	
Jumping	99.7	99.8	99.9	99.2	99.8	99.8	99.7	99.8	99.8	99.9	99.8	99.8	99.9	99.9	99.8	
Laying	95.7	92.1	89.4	96.8	96.7	98.2	94.6	97.04	89.3	97.5	98.2	97.5	98.4	97.4	98.2	

this sensor with neck accelerometer, the algorithm can distinguish each of the discussed activities.

Because of situation like this, we decided to compare the results using different number of accelerometers and different body placements. The idea is to use as few sensors as possible to maximize the user's comfort, but to use enough of them to achieve satisfactory performance.

Classified as

Activity	1	2	3	4	5	6	7	8	9	10	11
1	73.6	0.0	1.3	0.7	0.9	0.4	12.2	0.0	0.0	0.0	11.0
2	0.0	77.7	0.0	1.6	0.4	0.4	4.3	0.0	0.0	0.0	15.6
3	0.3	0.0	75.0	3.5	1.2	0.0	7.8	0.0	0.0	0.0	12.2
4	0.0	0.0	3.3	74.8	1.5	0.5	9.7	0.0	0.0	0.0	10.3
5	0.4	1.5	2.1	5.4	71.4	0.0	9.3	0.0	0.0	0.0	10.0
6	0.0	0.0	0.0	0.0	0.1	98.6	1.3	0.0	0.0	0.0	0.0
7	0.1	0.1	0.0	0.0	0.0	0.4	97.1	1.9	0.3	0.0	0.2
8	0.0	0.0	0.0	0.0	0.0	0.0	11.0	84.5	0.1	0.0	4.5
9	0.0	0.0	0.0	0.0	0.0	0.5	10.6	1.0	88.0	0.0	0.0
10	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	99.8	0.0
11	0.1	0.1	0.2	0.1	0.2	0.0	0.7	0.3	0.0	0.0	98.2

Figure 3: Confusion matrix for Neck and Ankle Accelerometer

We must make a trade-off between correctly detecting simple activity and specific fall. The results showed that neck and ankle accelerometers are best suited for fall detection with overall accuracy of 94.19%. The confusion matrix for neck and ankle accelerometers is shown in Figure 3. The most false positive predictions for fall activities are predicted as laying. Also, very small percent of the non-fall activities are predicted as falls, which dismiss the false alarms for falls.

5 CONCLUSION

In this paper we presented an approach to human activity recognition and how location and number of sensors can impact on the process of HAR. Our aim was to build a model who can correctly recognize and classify the fall activities using small number of accelerometers, but still can obtain high accuracy scores. With one accelerometer placed on the ankle or the neck we got high accuracy scores, but by combining these two sensors the model can classify the falls more precisely.

The main input to our system is the data from the inertial sensors. Because the data is sensory, additional attributes are calculated. This process of feature extraction is general and can be used in similar problems. Next, the algorithms for the final tasks of activity recognition and fall detection are designed and implemented using the data from the ankle accelerometer. We used a machine learning approach for solving the problem of activity recognition. We evaluated the

models and Random Forest showed best results. Then, we compared the best model on different data, and we got the conclusion that the data from ankle and neck sensors was sufficient for human activity recognition and fall detection process with accuracy of 94.2%.

REFERENCES

- [1] United Nations Publications. World Population Ageing 2019 Highlights. Department of Economic and Social Affairs Population Division.
- [2] Labrador, Miguel A., and Oscar D. Lara Yejas. Human activity recognition: using wearable sensors and smartphones. CRC Press, 2013.
- [3] Ravi, N., Dandekar, N., Mysore, P. and Littman, M.L., 2005, July. Activity recognition from accelerometer data. In Aaai (Vol. 5, No. 2005, pp. 1541-1546).
- [4] Kwapisz, J.R., Weiss, G.M. and Moore, S.A., 2011. Activity recognition using cell phone accelerometers. ACM SigKDD Explorations Newsletter, 12(2), pp.74-82.
- [5] Gjoreski, H., Lustrek, M. and Gams, M., 2011, July. Accelerometer placement for posture recognition and fall detection. In 2011 Seventh International Conference on Intelligent Environments (pp. 47-54). IEEE.
- [6] Gjoreski, M., Gjoreski, H., Luštrek, M. and Gams, M., 2016. How accurately can your wrist device recognize daily activities and detect falls?. Sensors, 16(6), p.800.
- [7] Atallah, L., Lo, B., King, R. and Yang, G.Z., 2011. Sensor positioning for activity recognition using wearable accelerometers. IEEE transactions on biomedical circuits and systems, 5(4), pp.320-329.
- [8] Bonomi, A.G., Plasqui, G., Goris, A.H. and Westerterp, K.R., 2009. Improving assessment of daily energy expenditure by identifying types of physical activity with a single accelerometer. Journal of applied physiology.
- [9] The Challenge UP dataset: <http://sites.google.com/up.edu.mx/har-up/>
- [10] Igual, R., Medrano, C. and Plaza, I., 2013. Challenges, issues and trends in fall detection systems. Biomedical engineering online, 12(1), p.66.
- [11] Z Zhang, C Conly, V Athitsos. 2015. A survey on vision-based fall detection. 8th ACM International Conference on PETRA '15, ACM, New York, NY, USA, Article 46, 1–7.
- [12] Attal, F., Mohammed, S., Dedabrishvili, M., Chamroukhi, F., Oukhellou, L. and Amirat, Y., 2015. Physical human activity recognition using wearable sensors. Sensors, 15(12), pp.31314-31338.
- [13] Altun, K., Barshan, B. and Tunçel, O., 2010. Comparative study on classifying human activities with miniature inertial and magnetic sensors. Pattern Recognition, 43(10), pp.3605-3620.
- [14] M. Gjoreski, V. Janko, G. Slapničar, M. Mlakar, N. Reščič, J. Bizjak, V. Drobnič, M. Marinko, N. Mlakar, M. Luštrek, M. Gams, Classical and deep learning methods for recognizing human activities and modes of transportation with smartphone sensors, Information Fusion, Volume 62, 2020, Pages 47-62, 1566-2535.

Sistem za ocenjevanje esejev na podlagi koherence in semantične skladnosti

Automated Essay Evaluation System Based on Coherence and Semantic Consistency

Žiga Simončič

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko

Večna pot 113, 1000 Ljubljana
zs3179@student.uni-lj.si

Zoran Bosnić

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko

Večna pot 113, 1000 Ljubljana
zoran.bosnic@fri.uni-lj.si

POVZETEK

V članku opisujemo implementacijo sistema za ocenjevanje esejev v angleškem jeziku. Zgledujemo se po metodologiji obstoječega sistema, ki poleg ocenjevanja sintakse uporablja tudi mere koherentnosti in semantične skladnosti. Metodologijo implementiramo v grafičnem okolju Orange, s prijaznim vmesnikom, opcijsko uporabo vektorskih vložitev za predstavitev besedila in možnostjo nadaljnega razvoja sistema. Sistem evalviramo na podatkih dostopnih na spletnem mestu Kaggle in, kolikor je mogoče, rezultate primerjamo z rezultati dosedanje metodologije in jih podrobno analiziramo. Poglobimo se tudi v izbiranje atributov za izboljšanje rezultatov. Glavni prispevki dela obsegajo (1) implementacijo sistema, (2) enostavnost uporabe in (3) izboljšave dosedanjega dela, vključno z dodatnimi računskimi opcijami in podrobno analizo izbiranja atributov za izboljšanje rezultatov.

KLJUČNE BESEDE

ocenjevanje esejev, semantična skladnost, Orange

ABSTRACT

In this paper we describe an implementation of an essay grading system. We lean heavily on the methodology of an existing system, which, besides using syntactical measurements, also uses coherence and semantic consistency measures. We implement the methodology in the Orange data mining tool, with a friendly user interface, optional use of word embeddings for word representation and the possibility for further developments of the system. The system is evaluated on public datasets from the Kaggle website. The results are to the most possible extent compared with the results of the existing methodology and analyzed in detail. We also compare several attribute selection methods, which improve our results. Main contributions of this work are comprised of (1) implementation of the system, (2) ease of use and (3) improvements upon previous work, including additional computing options and detailed attribute selection analysis.

KEYWORDS

automated essay evaluation, semantic consistency, Orange

1 UVOD

Učitelji v izobraževalnih ustanovah so odgovorni za predajanje znanj velikemu številu učencev. Del učnega procesa je tudi pisanje esejev, ki jih morajo učitelji prebrati in oceniti. Ocenjevanje esejev ni le časovno potratno, ampak potencialno tudi nekoliko pristransko. Naloga učitelja je tudi, da napake označi, popravi in komentira celotno delo.

S pomočjo računalnika lahko ocenjevanje esejev olajšamo. Dandanašnji sistemi za ocenjevanje esejev (tudi komercialni) se

osredotočajo predvsem na sintaksno analizo, premalo pozornosti pa posvečajo semantiki [6]. To slabost obstoječih sistemov rešuje sistem SAGE, ki ga Zupanc opisuje v svoji disertaciji [5]. SAGE dosega zavidljivo napovedno točnost v primerjavi z ostalimi sodobnimi sistemi, vendar je trenutna implementacija sistema v prototipni fazi in ni zrela za produkcijo.

Glavni cilj dela je bila implementacija sistema na način, da bo uporabnikom čimbolj dostopen, enostaven in prijazen za uporabo. Da zadostimo tem ciljem, smo se odločili za implementacijo v programskem okolju Orange,¹ ki je namenjen hitremu prototipiranju modelov in raziskovanju podatkov, namenjen tako začetnikom kot zahtevnejšim uporabnikom. Sistem je v Orange-u implementiran v obliki gradnikov (angl. widgets). Med seboj jih lahko povezujemo in kombiniramo, tako da smo uvoz datotek, gradnjo in testiranje modelov prepustili gradnikom, ki so v Orange-u že implementirani. Skupno smo implementirali tri gradnike – prvi implementira vse atributske funkcije, vključno s koherenco, drugi implementira sistem za analizo semantične skladnosti, tretji pa je namenjen evalvaciji modela po kvadratno uteženi kapi.

Sistem Zupanc [6] temelji na ekstrakciji različnih atributov iz podanih besedil (esejev) in se loči na tri (pod)sisteme: *AGE*, *AGE+* in *SAGE*. Oznaka "sistem Zupanc" predstavlja njeno implementacijo vseh teh treh sistemov. Vsak sistem nadgradi prejšnjega z dodatnimi atributi. Sistem *AGE* predstavlja skupek atributov osnovne sintaktične statistike, berljivostnih, leksikalnih, slovničnih in vsebinskih mer. To obsega različne značilnosti besedila, vse od osnovnih, kot so število znakov, besed itd., pa do števila slovničnih napak in računanje podobnosti z ostalimi esei. Skupno ta sistem zajema 72 različnih atributov, v prispevku tega članka pa smo temu sistemu dodali še pet novih atributov (št. znakov brez presledkov in štiri dodatne attribute, ki štejejo število posameznih oblikoskladenjskih oznak). Skupno torej 77 atributov.

Atributom sistema *AGE* dodamo attribute za merjenje koherence in s tem dobimo sistem *AGE+*. Koherenco merimo tako, da besedilo najprej razdelimo na prekrivajoče se odseke (drseče okno) in posamezne odseke pretvorimo v večdimenzionalni prostor. V tem prostor lahko posamezne odseke primerjamo in z različnimi merami ocenimo konsistentnost besedila in tok misli. Število atributov za merjenje koherence je 29.

Če vsem zgornjim atributom dodamo še nabor treh atributov, ki jih pridobimo s preverjanjem semantične skladnosti, govorimo o sistemu *SAGE*. Sistem za zaznavanje semantičnih napak v ozadju uporablja ontologijo, kateri postopoma dodajamo dejstva, ki jih izluščimo iz besedila. Z logičnim sklepanjem nato ugotovimo, če so trditve iz besedila logično konsistentne ali ne. To nam prinese tri dodatne attribute in možnost povratne informacije, v katerih povedih je prišlo do semantičnega neskladja.

¹<https://orange.biolab.si/>

2 SORODNA DELA

V sklopu svojega dela se je Zupanc [5] osredotočila na (v času njenega raziskovanja že zaključeno) tekmovanje avtomatskega ocenjevanje esejev, ki ga je gostil Kaggle.² Na tem tekmovanju so pomerili različni sistemi, s katerimi je Zupanc primerjala svoj sistem. Najboljša mesta na končni lestvici so večinoma zasedali komercialni sistemi za ocenjevanje esejev, nekaj pa je bilo tudi po meri narejenih uporabniških modelov. Komercialni sistemi kot so PEG,³ e-rater⁴ in IntelliMetric⁵ imajo že dolgo zgodovino in s tem velik tržni delež ter izpopolnjen finančni model. V času raziskovanja noben od naštetih ni ponujal brezplačne verzije sistema. Podrobno razčlenitev modelov in splošen opis njihovega delovanja najdemo v delih Zupanc [5] ter Zupanc in Bosnić [6].

V zadnjem času se na različnih področjih čedalje bolj uveljavljajo nevronske modeli, zato smo pogledali in testirali nekaj izvedb. Martinc in sod. [3] opisujejo uspešnost treh različnih nevronske modelov pri ocenjevanju besedil, ki sicer niso eseji. Tudi Taghipour in Tou Ng [4] sta primerjala različne nevronske modele za ocenjevanje esejev (na istih podatkih kot mi). Najboljši model dosega skoraj tak rezultat, kot mi. Alikaniotis in sod. so objavili članek [1], kjer so tudi testirali uspešnost različnih nevronske modelov na enaki podatkovni zbirki esejev, kot smo jo uporabljali mi.

3 OPIS IMPLEMENTACIJE IN METODE

3.1 Uporabljena orodja

Celoten sistem smo implementirali z uporabo orodja za podatkovno rudarjenje Orange v programskem jeziku Python. Glavne uporabljene knjižnice za razčlenitev besedila in izračun atributov so NLTK,⁶ SpaCy,⁷ scikit-learn⁸ in language-check⁹ za zaznavanje pravopisnih napak.

Za delo z ontologijami smo uporabili knjižnico rdflib¹⁰ in zunanja sistema (v smislu samostojna lokalna programa) ClausIE (na voljo tudi OpenIE5.0) in Hermit.¹¹

3.2 Implementacija gradnikov v Orange

Skupno smo razvili tri gradnike, ki zajemajo celoten opisan sistem. Slika 1 prikazuje vse tri gradnike, ki so opisani v nadaljevanju.

Prvi gradnik je namenjen izračunu vseh različnih mer. To so osnovne (plitke) statistične mere, mere berljivosti, leksikalne mere, slovnične mere, vsebinske mere in mere koherentnosti. Gradnik predstavlja sistema AGE in AGE+, odvisno od uporabnikove izbire atributov, ki naj se izračunajo. Če označimo izračun vseh atributov, razen atributov za koherenco, govorimo o sistemu AGE, z dodanimi atributi za koherenco pa govorimo o sistemu AGE+. Ker je računanje nekaterih naprednih mer bolj zahtevno, se lahko uporabnik odloči za izračun kakršnekoli kombinacije naštetih šestih skupin mer. Za vsebinske mere in mere koherentnosti je na voljo dodatna izbira metode pretvorbe besedila v večdimenzionalni vektorski prostor. Tu podpiramo dve metodi: statistično pretvorbo TF-IDF in vektorske vložitve GloVe (v dveh izvedbah: SpaCy in Flair).

²<https://www.kaggle.com/>

³<https://www.measurementinc.com/products-services/automated-essay-scoring>

⁴<https://www.ets.org/>

⁵<http://www.intellimetric.com/direct/>

⁶<https://www.nltk.org/>

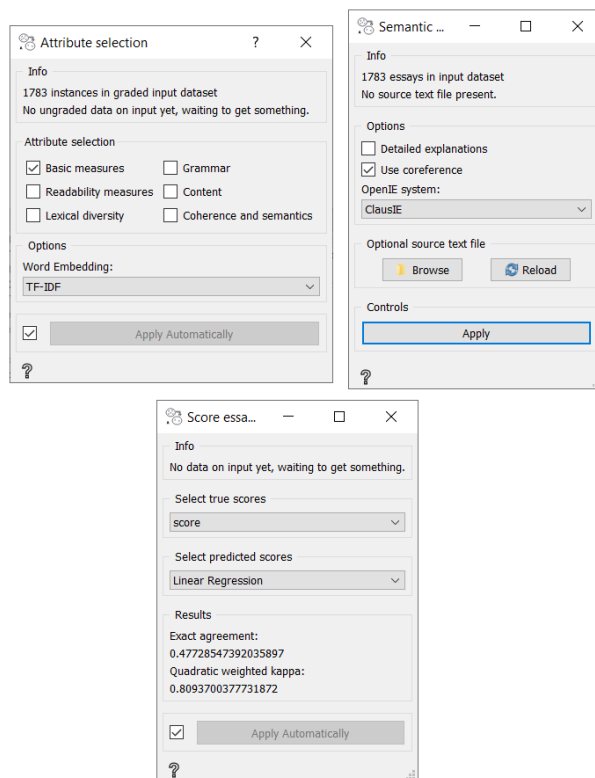
⁷<https://spacy.io/>

⁸<https://scikit-learn.org/stable/>

⁹<https://pypi.org/project/language-check/>

¹⁰<https://rdflib.readthedocs.io/en/stable/>

¹¹<http://www.hermit-reasoner.com/>



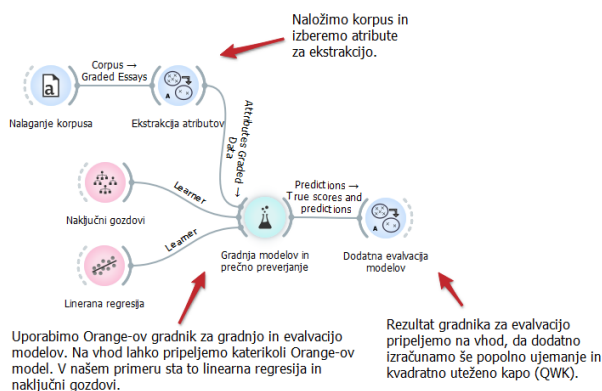
Slika 1: Prikaz vseh treh gradnikov

Gradnik ima tri vhode:

- (1) vhod za ocenjene eseje,
- (2) vhod za neocenjene eseje in
- (3) vhod za izvorno besedilo.

Vhoda za ocenjene in neocenjene eseje sta namenjena učni množici ocenjenih esejev in množici neocenjenih esejev, ki jim hočemo napovedati ocene. Na obeh množicah se izračunajo enaki atributi. Attribute ocenjenih esejev uporabimo za gradnjo modela. Vhod za izvorno besedilo je neobvezen in predstavlja izhodiščno zgodbo, knjigo ali dejstva, ki naj bi jih pisec esejev poznal. Če so eseji osnovani na podlagi nekega izvornega besedila, ga povežemo na ustrezen vhod in s tem izračunamo dodaten atribut (podobnost eseja z izvornim besedilom). Gradnik ima dva izhoda, in sicer izhod za izračunane attribute ocenjenih esejev in izhod za izračunane attribute neocenjenih esejev. To nam omogoča, da podatke ustrezno nastavimo kot vhode v ostale Orange-ove gradnike.

Drugi gradnik obsega delo in iskanje semantičnih neskladnosti z ontologijo. Predstavlja izračun dodatnih atributov, ki jih prinaša sistem SAGE. Gradnik je samostojen zaradi velike računske in časovne zahtevnosti. Ima dve nastavitvi: ali želimo uporabiti razreševalnik koreferenc in ali želimo, da se nam za semantične napake vrne podrobna razlaga. Uporaba koreferenc je priporočljiva, saj je v primerih posrednega navezovanja na različne pojme v besedilu to edini način zajetja celotne semantične informacije. Izberemo lahko tudi izvorno besedilo ali zgodbo, s katerim se razširi ontologijo, tako da ta vključuje tudi vsebino osnovnega besedila. To besedilo se bo obdelalo pred vsem ostalim, izluščene trojice pa bodo dodane v ontologijo. Razširjena ontologija se bo uporabila za preverjanje skladnosti esejev. Če



Slika 2: Primer uporabe sistema AGE/AGE+

izvornega besedila ne dodamo, se za preverjanje skladnosti normalno uporabi osnovna ontologija (ontologija COSMO). Gradnik ima samo en vhod — vhod za eseje ter en izhod — tabela treh atributov o številu posameznih napak in niz z osnovno razlago ter dodatni stolpec s podrobno razlago, če je ta izbrana.

Tretji gradnik je namenjen evalvaciji napovedanih ocen in pravih ocen esejev. Ker Orange ne podpira mer za izračun natančnega strinjanja (angl. *exact agreement*) in kvadratne utežene kape (angl. *quadratic weighted kappa - QWK*), smo naredili gradnik, ki prejme tabelo z napovedanimi ocenami in praviimi ocenami. Zgledovali smo se po izhodu gradnika *Test and Score* — za zagotavljanje interoperabilnosti lahko ta izhod vezemo neposredno na vhod našega gradnika, kjer se izračunata prej omenjeni meri.

Uporaba gradnika za izračun atributov in evalvacijo modela s kvadratno uteženo kapo je prikazana na Sliki 2.

3.3 Semantična analiza

Eden glavnih prispevkov dela Zupanc in Bosnić [6] je uporaba ontologij za ugotavljanje semantične skladnosti. Ta postopek je uporaben na dva načina: z njim pridobimo nekaj dodatnih atributov, ki jih lahko uporabimo pri napovedovanju ocen esejev, dodatno pa nam ta postopek tudi sporoči, kje se nahajajo semantične napake. Slednja funkcionalnost je zelo pomembna, saj tako učenec prejme neposredno informacijo o napakah v eseju.

Postopek temelji na uporabi ontologije, v katero postopoma dodajamo v relacije strukturirane stavke in sproti preverjamo skladnost ontologije. Osnovna struktura ontologije je predstavljena s "trojicami" v obliki (*osebek, relacija, predmet*). Relacija lahko predstavlja omejitev, konceptualno povezavo (npr. (*Alice, isMotherOf, Bob*)) ali definira tip. V implementaciji smo za predstavitev trojic uporabili jezik RDF, ki je podoben jeziku OWL, vendar ni logični jezik. Uporabili smo ontologijo COSMO (angl. *Common Semantic Model*). Predstavljena je v semantičnem jeziku OWL¹² (*Web Ontology Language*), ki omogoča gradnjo kompleksnih shem različnih konceptov, dejstev in medsebojnih relacij. V primeru, da bi hoteli ontologiji dodati dodatna specifična znanja, to lahko storimo. V našem primeru je poleg nekaterih esejev tudi izvorno besedilo, na podlagi katerega so bili eseji spisani. Izvorno besedilo dodamo v ontologijo pred eseji in po enakem postopku kot eseji in je razložen spodaj.

¹²<https://www.w3.org/OWL/>

Za posamezen esej poiščemo koreference v besedilu (angl. *co-reference resolution*). Ugotavljanje referenc nam omogoča odkrivanje posrednih referenc na določene entitete in zamenjavo z neposredno entiteto. Primer: "*Bob likes pizza. He eats it all the time.*" nadomestimo z "*Bob likes pizza. Bob eats pizza all the time.*"

Naslednji korak je razčlenitev besedila na posamezne povedi in ekstrakcija informacij s pomočjo sistema OpenIE (angl. *Open Information Extraction*). V tem koraku posamezne povedi pretvorimo v eno ali več trojic, ki opišejo relacije, izražene v povedi in so primerne za logično obdelavo. Za zgornji primer bi tako dobili dve trojici: (*Bob, like, pizza*) in (*Bob, eat, pizza*). Uporabili smo sistem za ekstrakcijo ClausIE [2], podpiramo pa tudi možnost uporabe sistema OpenIE5.¹³ Vse pridobljene trojice nato postopoma dodajamo v ontologijo, obenem pa preverjamo njeno skladnost. Za vsak element trojice poskušamo v ontologiji najti že obstoječ element. Pri tem preiščemo sopomenke, nadpomenke in protipomenke, v najslabšem primeru pa dodamo v ontologijo nov element. Po vsakem dodajanju elementov in trojic, preverimo skladnost ontologije. Skladnost preverjamo z logičnim sklepalnikom HermiT, ki vrača dva tipa napak. Prvi tip napak se zgodi, ko ima nek razred (*owl:Class*) prirajene entitete, ki jih ne sme imeti (*unsatisfiable case*). Drugi tip napak pa se proži, ko se s sklepanjem ugotovi logična napaka — nekonsistentna ontologija (angl. *inconsistent ontology*). Do takšnih napak pride ponavadi zaradi neposrednih nasprotij (npr. *owl:disjointWith*) med dvema relacijama, ki pravi, da entiteta ne more imeti obeh relacij hkrati.

Na podlagi povzročenih tipov napak osnujemo tri dodatne attribute, ki jih lahko uporabimo pri napovedovanju ocen esejev: število neizpolnjenih primerov (pri dodajanju novih entitet v ontologijo), število napak nekonsistentne ontologije (pri dodajanju trojic) in vsota obeh prejšnjih.

3.4 Rezultati

Sistem smo testirali na podatkih že nekaj let starega tekmovanja ASAP na spletni strani Kaggle.¹⁴ Podatki obsegajo osem različnih podatkovnih zbirk (oz. devet, ker se druga zbirka ocenjuje po dveh kriterijih). Tema esejev v vsaki podatkovni zbirki je različna. Zbirke so razdeljene na učno, validacijsko in testno množico, vendar ocene validacijske in testne množice niso na voljo, zato smo za evalvacijo našega sistema uporabili 10-kratno prečno preverjanje. Razpon ocen je v vsaki zbirki različen, gibljejo se od 0–4, pa vse do 0–60. Za oceno modelov smo uporabili mero kvadratno utežene kape (angl. *quadratic weighted kappa*), ki upošteva razpon ocen in vrne relativno ujemanje napovedane ocene z dejansko oceno. Sistem smo testirali na modelu linearne regresije in naključnih gozdov. Bolje se je odrezala linearna regresija, zato smo se nanjo osredotočili v nadaljnjih eksperimentih. Uporabili smo regularizacijo L2 s parametrom $\alpha = 0,02$.

Na začetku smo modele gradili na celotnem naboru izračunanih atributov. Ker sistem AGE+, domnevno zaradi prevelikega števila atributov (106), ni dosegal boljših rezultatov od sistema AGE, smo preizkusili nekaj metod za izbiranje atributov. Glavni metodi naše analize sta bili vnaprejšnje izbiranje atributov (angl. *forward attribute selection*) in izločanje atributov (angl. *backward feature elimination*). Obe metodi sta izboljšali rezultat. Uporabili smo jih skupaj z 10-prečnim preverjanjem. Na vsaki iteraciji prečnega preverjanja smo dodali/odstranili posamezne attribute in glede na povprečje preko vseh iteracij dodali/odstranili atribut z največjim/najmanjšim prispevkom. To smo ponavljali, dokler ni bilo

¹³<https://github.com/dair-iitd/OpenIE-standalone>

¹⁴<https://www.kaggle.com/c/asap-aes>

Tabela 1: Primerjava rezultatov brez izbiranja atributov naše implementacije sistemov AGE in AGE+ (TF-IDF), primerjava s sistemom Zupanc (AGE) in strnjeni rezultati izbiranja ter izločanja atributov na sistemu AGE+

	Brez izbiranja			Izbiranje	Izločanje
	AGE	AGE+	Zupanc (AGE)		
DS1	0,8358	0,8343	0,8447	0,8369	0,8439
DS2a	0,7001	0,7073	0,7389	0,7158	0,7324
DS2b	0,6789	0,6676	0,5386	0,6941	0,7028
DS3	0,6578	0,6622	0,6591	0,6656	0,6958
DS4	0,7536	0,7547	0,7174	0,7619	0,7769
DS5	0,7964	0,7955	0,7949	0,8028	0,8122
DS6	0,7734	0,7675	0,7636	0,7771	0,7871
DS7	0,8071	0,8034	0,7888	0,8083	0,8183
DS8	0,7479	0,7428	0,7738	0,7681	0,7717
AVG	0,7501	0,7484	0,7356	0,759	0,7712

več izboljšanja. Pri analizi smo opazili, da je nabor atributov, ki pride v končni izbor, relativno majhen. Ugotovili smo, da je zaradi prečnega preverjanja velika možnost, da s trenutnim naborom atributov pridemo v lokalni optimum. Zaradi povprečenja čez vse iteracije lahko nek atribut v prvi iteraciji izboljša rezultat, v drugi pa poslabša, in je v povprečju označen kot neprimeren. Za izogibanje tem lokalnim optimumom smo implementirali mejo, kolikokrat se lahko v povprečju rezultat poslabša, preden nabor atributov označimo kot končen. S tem smo kratkoročno poslabšali rezultat, vendar dolgoročno ustvarili kombinacijo atributov, ki dajejo v povprečju boljši rezultat. S to metodo izogibanja optimumov smo še dodatno izboljšali končne rezultate, ki so strnjeno prikazani v Tabeli 1. Pri izbiranju in izločanju atributov je AGE izpuščen, saj AGE+ v obeh primerih dosega boljše rezultate. Ker testni podatki niso več na voljo, smo naše rezultate s sistemom Zupanc lahko primerjali le s primerjavo sistemov AGE z 10-kratnim prečnim preverjanjem. Vidimo, da dosegamo zelo podobne rezultate, kot sistem Zupanc oz. jih nekoliko presehamo. Z ustreznim izbiranjem atributov pa naš rezultat še dodatno izboljšamo.

Sistem SAGE smo iz tabele izpustili, saj so rezultati z izločanjem atributom le malenkost boljši od sistema AGE+, prav tako pa smo ga uporabili le na podatkovnih zbirkah, ki so vsebovale izvorno besedilo (samo štiri zbirke). Kljub temu pa sistem SAGE ob zaznanem semantičnem neskladju nudi izpis povratne informacije. Primer v nadaljevanju prikazuje delovanje razreševalnika koreferenc in odkrivanje semantičnih napak. Zaradi korenjenja so nekatere besede v razlagi lahko odsekane. Vhod "George likes basketball and doesn't like sports.", sproži napako z razlago: "Relation 'George likes basketball and George doesn't like sports.' is inconsistent with a relation in ontology: 'George likes basketball and George doesn't like sports.'" in podrobno razlago: "Relation not consistent: Georg likes Basketball. Relations doesNotLike and likes are opposite/disjoint. Relation not consistent: Georg doesNotLike Basketball.". Osnovna razlaga deluje na ravni povedi in nam v tem primeru pove, da je poved v nasprotju sama s sabo. Podrobna razlaga pravi, da George ima in nima rad košarke. Beseda "sports" se v podrobni razlagi ne pojavi, ker je košarka podrazred športa in tam najprej pride do nasprotja.

Omenili bi še primerjavo našega sistema z omenjenimi nevronske modeli. Model Taghipour in Tou Ng [4] dosega podobne rezultate, kot naš sistem (nekaj pod 0,77). Alikaniotis in sod. [1] opisujejo, da njihov model dosega rezultat 0,96, vendar sumimo na nekaj nepravilnosti, ki izvirajo iz napačne uporabe mere za

ocenjevanje modelov (kvadratno utežene kape). Sumimo, da so za učenje svojega modela uporabili vse podatkovne zbirke skupaj, saj je njihov rezultat v območju skoraj 100% natančnosti (0,96), z dvakrat večjo absolutno napako (RMSE), kot naš model, ki ima rezultat približno 0,77. Z uporabo vseh zbirk na našem sistemu tudi dobimo tako visok rezultat (0,97 in 0,94, odvisno od modela).

4 ZAKLJUČEK

V sklopu tega dela smo implementirali sistem za ocenjevanje esejev po zgledu dela Zupanc [5] v programskem okolju Orange. Implementacija v okolju Orange omogoča enostavno uporabo sistema in združljivost z že implementiranimi funkcionalnostmi Orange-a. Sistemu smo dodali nekaj novih atributov in možnost predstavitve besed z vektorskimi vložitvami GloVe. Naša implementacija sistema je na voljo na repozitoriju git.¹⁵ Sistem temelji na ekstrakciji velikega števila atributov iz besedil in nato izboru najboljšega nabora za določeno podatkovno zbirko. Inovativni del preteklega dela, ki je vključen tudi v naši implementaciji, je dodaten sistem za preverjanje semantične skladnosti, s pomočjo katerega nabor atributov dodatno obogatimo, obenem pa imamo možnost, da nam sistem izpiše vse zaznane semantične napake oz. neskladja. Prispevek tega članka predstavlja tudi primerjava tehnik izbiranja atributov in primerjava rezultatov s preteklim delom. Sistem bi bilo smiselno preizkusiti tudi z drugimi napovednimi modeli, saj smo se v našem delu najbolj osredotočili le na linearno regresijo in naključne gozdove. Zanimiv izziv bi bil tudi prilagoditev sistema za slovenski jezik, ker je jezik sintaktično kompleksnejši, orodja za obdelavo besedil pa še niso tako zrela kot za angleški jezik.

ZAHVALA

Zahvaljujemo se sodelavcem Laboratorija za bioinformatiko na Fakulteti za računalništvo in informatiko za podporo in nasvete pri implementaciji sistema v programskem okolju Orange.

LITERATURA

- [1] Dimitrios Alikaniotis, Helen Yannakoudakis in Marek Rei. 2016. Automatic text scoring using neural networks. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. doi: 10.18653/v1/p16-1068. <http://dx.doi.org/10.18653/v1/P16-1068>.
- [2] Luciano Del Corro in Rainer Gemulla. 2013. ClausIE: Clause-Based Open Information Extraction. V *Proceedings of the 22nd international conference on World Wide Web*, 355–366.
- [3] Matej Martinc, Senja Pollak in Marko Robnik-Šikonja. 2019. Supervised and unsupervised neural approaches to text readability. *arXiv preprint arXiv:1907.11779*.
- [4] Kaveh Taghipour in Hwee Tou Ng. 2016. A neural approach to automated essay scoring. V *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Austin, Texas, (november 2016), 1882–1891. doi: 10.18653/v1/D16-1193. <https://www.aclweb.org/anthology/D16-1193>.
- [5] Kaja Zupanc. 2018. Semantics-based automated essay evaluation. Doktorska disertacija. Fakulteta za računalništvo in informatiko, Univerza v Ljubljani.
- [6] Kaja Zupanc in Zoran Bosnić. 2017. Automated essay evaluation with semantic analysis. *Knowledge-Based Systems*, 120, 118–132.

¹⁵<https://github.com/venom1270/essay-grading>

Mental State Estimation of People with PIMD using Physiological Signals

Gašper Slapničar

gasper.slapnicar@ijs.si

Jožef Stefan Institute, Jožef Stefan IPS

Jamova cesta 39

Ljubljana, Slovenia

Jakob Valič

jakob.valic@ijs.si

Jožef Stefan Institute

Jamova cesta 39

Ljubljana, Slovenia

Erik Dovgan

erik.dovgan@ijs.si

Jožef Stefan Institute

Jamova cesta 39

Ljubljana, Slovenia

Mitja Luštrek

mitja.lustrek@ijs.si

Jožef Stefan Institute

Jamova cesta 39

Ljubljana, Slovenia

ABSTRACT

People with profound intellectual and multiple disabilities are a very diverse and vulnerable group of people. Their disabilities are cognitive, motor and sensory, and they are also incapable of symbolic communication, making them heavily reliant on caregivers. We investigated the connection between physiological signals and inner states as well as communication attempts of people with PIMD, using signal processing and machine learning techniques. The inner states were annotated by expert caregivers, and several heart rate variability features were computed from photoplethysmogram. We then fed the features into hyper-parameter-tuned classification models. We achieved the highest accuracy of 62% and F1-score of 0.59 for inner state (pleasure, displeasure, neutral) classification using Extreme Gradient Boosting, which notably surpassed the baseline.

KEYWORDS

PIMD, mental state, physiological signals, classification

1 INTRODUCTION

People with profound intellectual and multiple disabilities (PIMD) often face extreme difficulties in their day-to-day life due to severe cognitive, motor and sensory disabilities. They require a nearly everpresent caregiver to help them with most tasks. Additionally, they are unable to communicate their feelings or express their current mental state in a traditional symbolic way. This causes a gap between a caregiver and the care recipient, as it can take an extended period of time for the caregiver to recognize any potential patterns and their relationship with the mental state of the care recipient.

The aforementioned reasons call for a technological solution that might help bridge the gap between the caregiver and the care recipient and help the former better understand the latter. The INSENSATION project [8] aims to develop such assistive technology, which takes into account many aspects of the care recipient. The aim is to both bridge the previously mentioned gap as well as empower the people with PIMD to be able to interact with

their surroundings through technology. One part of the system considers the patterns in a person's gestures and facial expressions, which might have some significance and correlation to their behavioural and mental state, or their communication attempt. The initial solution dealing with this part was already described by Cigale et al. [1, 2]. In this paper, we instead focus on exploring the relationship between the physiological response of the body and the mental state of the people with PIMD by using features computed from photoplethysmogram (PPG). PPG is a periodic signal, where each cycle corresponds to a single heart beat. We obtained the PPG in two different ways: 1.) by using a high-quality wearable Empatica E4 with an optical sensor measuring the reflection of light from the skin and 2.) by using a contact-free RGB camera mounted on a wall, which records the color changes of the skin pixels. The features were then used to train classification models, which predicted the person's inner state or communication attempt.

The rest of this paper is structured as follows: we first investigate the related work in Section 2, then we describe the data collected and used in the experiments in Section 3. We continue with the methodology and experimental setup description in Section 4, and conclude with results and discussion in Section 5.

2 RELATED WORK

The connection between physiological parameters and mental states is a mature and highly-researched field when it comes to average healthy people.

Schachter et al. [6] investigated the emotional state of people as a function of cognitive, social and physiological state. Several propositions were made and experimentally confirmed, supporting the overall connection between emotional and physiological state.

Cigale et al. [1, 2] explored the communication signals of people with PIMD, which are atypical and idiosyncratic. They highlighted the challenging interpretation of these signals and their meaning and suggested how technology could help overcome the gap between caregivers and care recipients. Some models were proposed that take the person's non-verbal signals (NVS) as input and classify their inner state or communication attempt.

Kramer et al. [3] highlighted the challenges of analysing the NVS in people with PIMD, as they are difficult to discern, instead focusing on physiological body responses. They conducted a research in which the expressions of three emotional states of one person with PIMD were recorded during nine emotion-triggering

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

Information society '20, October 5–9, 2020, Ljubljana, Slovenia

© 2020 Copyright held by the owner/author(s).

situations. They collected heart rate (HR) and skin conductance level (SCL), and investigated the connection between these two physiological signals and the emotional state. They found higher SCL activity during anger or happiness and lower SCL activity during relaxation or neutral state.

Vos et al. confirmed that HR and skin temperature allow the same conclusions in people with PIMD and people without disabilities, regarding positive and negative emotion. This finding gives additional motivation to our work, showing that the connection between physiological and mental state also holds for people with PIMD [9].

3 DATA

We created a recording setup in the INSENSION project, which uses two Logitech C920 cameras capable of recording full HD (1920x1080) resolution video at 30 frames per second (fps). The cameras were setup perpendicular to one another to record from two distinct angles, allowing for decent facial exposure even when the face changes direction. The caregivers were instructed to attempt to conduct their activity in front of one of the cameras whenever possible. Additionally, the subjects were given an Empatica E4 wristband, which served both as the ground truth for PPG, as well as a fall-back mechanism for obtaining physiological signals in cases when camera is unreliable or unavailable. The wristband records PPG at 64 Hz, allowing for capture of reasonable morphological details. The temporal synchronization between the video and ground truth was ensured to the best of our abilities using suitable protocols and checks.

With the described setup, we obtained 48 recording sessions, each lasting between 10 and 30 minutes. Five sessions were eliminated immediately, as there was a large mismatch between the duration of the video and the duration of the ground truth, which may happen due to several reasons, such as a caregiver forgetting to turn on the wristband during a session or the wristband losing connection.

It is important to note that the recordings were made in a natural way, as the caregivers were not given any additional restrictions other than to be in front of the camera when possible. In practice this means that large parts of some recordings might be useless due to the person with PIMD being turned away or the caregiver blocking them. Examples of good and bad sessions are shown in Figure 1.

3.1 Annotating the ground truth

In order to classify mental states of people with PIMD, we first required the ground truth annotations. As it is generally difficult to obtain such ground truth, we relied on the expert knowledge of partners in the project who specialize in education of people with special needs, alongside the caregivers, who know their care recipients the best. Together they devised an annotation schema, in which they annotated inner states and communication attempts of people with PIMD and can take the values given in Equations 1 and 2.

$$InnerState = \begin{cases} displeasure & \text{if 1, 2 or 3} \\ neutral & \text{if 4, 5 or 6} \\ pleasure & \text{if 7, 8 or 9} \end{cases} \quad (1)$$

The three numbers within each mental state indicate the intensity, where a lower number for displeasure means more intense displeasure, and a higher number for pleasure indicates more intense pleasure.



Figure 1: Example of good (green) and bad (red) video recordings.

$$CommAttempt = \begin{cases} protest \\ comment \\ demand \end{cases} \quad (2)$$

The caregivers were tasked with annotation of videos, looking at camera recordings and marking inner states and communication attempts in time, always marking the start and end of each recognized state, regardless of duration (can be a few seconds or a few minutes). Naturally, large periods remained where nothing was annotated, as the experts were either not sure or did not recognize any of the pre-defined states. This does not mean that nothing is happening in those periods, but simply that the inner experience of the person with PIMD is unknown. Thus, we added an additional class value for the areas where nothing was annotated – unknown.

4 METHODOLOGY OF MENTAL STATE ESTIMATION

Having both the ground truth annotations and physiological data and videos, we then investigated two approaches: 1.) we attempted to reconstruct PPG from the camera recordings in a contact-free manner and use the reconstructed rPPG (remote PPG) to calculate features and to classify inner state and communication attempt and 2.) we directly used the Empatica ground truth PPG to calculate features to be used in the same classification task.

4.1 Using rPPG Reconstruction

In order to obtain the remote PPG, we used a rather standard pipeline, which was updated with a convolutional neural network in order to further enhance the rPPG. At a high level, the pipeline consists of detection or region of interest (ROI), extraction of red, green and blue signal components (RGB), detrending and band-pass filtering of RGB, rPPG reconstruction using the Plane

Orthogonal to Skin (POS) algorithm, band-pass rPPG filtering (0.5 to 4.0 Hz), and rPPG enhancement via deep learning. Details were already described in our previous work [7] and are not subject of this paper.

We ran the pipeline described above on 30-second segments of video using a sliding window without overlap. We decided to use 30 seconds due to the nature of some frequency features that we chose, as frequency analysis makes sense once a reasonable number of periods are available - in our case this means that a sufficient number of heart cycles must be available. Additionally, this length makes sense as we are primarily attempting to predict inner states, which do not change extremely in such a short time span. An example output of the pipeline is shown in Figure 2.

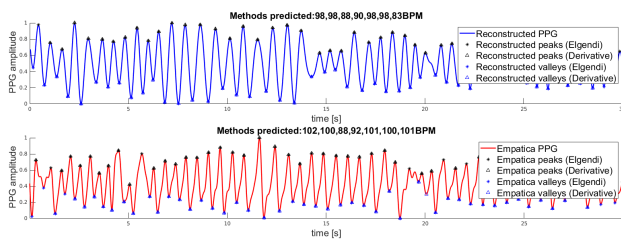


Figure 2: Example of a good rPPG segment obtained with our pipeline.

We then used the rPPG to compute several heart rate variability (HRV) features. These are known to be well-correlated with stress, cognitive load, conflict experience and other inner states [5, 4]. A detailed list of computed features is given in Table 1.

4.2 Using Empatica PPG

The Empatica records PPG directly on the skin, thus making the raw PPG readily available, without the need for additional reconstruction. Still, due to subject arm and wrist movements, we opted to use similar preprocessing steps used previously, namely detrending and band-pass filtering, as the signal can sometimes be quite noisy.

We computed the same set of features and window length as before (see Table 1), and used them in the same classification task, attempting to recognize inner states and communication attempts.

5 EXPERIMENTS AND RESULTS

Once both the input (HRV features) and output (annotations) were known, we investigated six classification algorithms (k Nearest Neighbours, Decision Trees, Random Forest, Support Vector Machines, AdaBoost and Extreme Gradient Boosting) for this task, always training separate models for inner state and communication attempt. We always compared each algorithm against a baseline majority vote classifier using two metrics, accuracy and F1-score.

5.1 Using Empatica PPG

We started our evaluation using the Empatica data, as it is more reliable, since the PPG reconstruction is not needed. At the time of evaluation, we had annotations for 15 recording sessions in which 2 different people with PIMD are present. Using the chosen 30-second window, we initially had 417 segments of Empatica PPG available. The unknown class label heavily skewed the data for both classes, and there is no way to know which (other) class

Table 1: List of computed HRV features.

Feature	Description
HRmean	$60/\text{mean}(NN)$
HRmedian	$60/\text{median}(NN)$
IBImedian	$\text{median}(NN)$
SDNN	$\text{std}(NN)$
SDSD	$\text{std}(\text{abs}(NN'))$
RMSSD	$\text{sqr}(\text{mean}((NN')^2))$
NN20 and NN50	The number of pairs of successive NNs that differ by more than 20ms and 50ms
pNN20 and pNN50	The proportion of NN20 and NN50 divided by total number of NNs
SDbonus1	$\text{sqr}(0.5) * \text{SDNN}$
SDbonus2	$\text{sqr}(\text{abs}(2 * \text{SDSD}^2 - 0.5 * \text{SDSD}^2))$
VLF	Area under periodogram in the very low frequencies
LF	Area under periodogram in the low frequencies
HF	Area under periodogram in the high frequencies
LFnorm and HFnorm	Area under periodogram in the low and high frequencies, normalized by the whole area under periodogram
LFdHF	LF/HF

where *std* is standard deviation,
abs is absolute value,
X' is the first order derivative,
sqr is the square root and
NN are the beat-to-beat intervals.

label it actually belongs to, so we decided to exclude it from evaluation. This left us with 272 instances for class inner state and 80 instances for class communication attempt, which was annotated more sparsely. The final distributions for each class are shown in Figure 3.

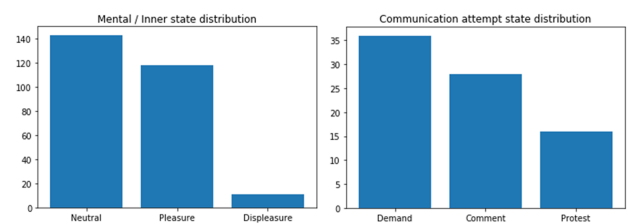


Figure 3: Distributions of both classes.

Initially we conducted a 5-fold cross validation (CV) to investigate the best hyper-parameters using a grid search. Once the hyper-parameters were determined, we ran a separate experiment, using the best overall hyper-parameters for each model. Again, we ran a 5-fold CV with the best hyper-parameter settings obtained on the full data to validate the performance. All the investigated algorithms (from the Scikit-learn and XGBoost packages) and their corresponding sets of optimized parameters with the best values are available from the authors, but are not listed here due to space restrictions. Results of our evaluation in terms of accuracy and F1-score for both classes are given in Table 2.

Table 2: Accuracy and F1 score for both classes.

Algorithm	$ACC_{mentalstate}$	$F1_{mentalstate}$
Baseline (majority)	0.52	0.36
kNN	0.55	0.55
Tree	0.54	0.56
RF	0.57	0.56
SVM	0.55	0.52
AdaBoost	0.59	0.56
XGB	0.62	0.59

Algorithm	$ACC_{commattemp}$	$F1_{commattemp}$
Baseline (majority)	0.45	0.27
kNN	0.42	0.42
Tree	0.41	0.39
RF	0.46	0.43
SVM	0.43	0.34
AdaBoost	0.43	0.41
XGB	0.48	0.45

5.2 Using rPPG reconstruction

Using the rPPG for evaluation proved to be more difficult, as we only had limited amount of good subsequent facial crops from the videos, while also having a limited amount of annotations. This meant that the overlap between the two was very small – we had only 12 such 30-second segments for inner state and only 6 for communication attempt. Such a low amount of data is infeasible to be used in a realistic evaluation scheme (not even all three different class labels were present), so we instead decided to use the models previously trained on the Empatica data, to classify these instances obtained via the rPPG. We achieved reasonably high accuracy of 75% and F1-score of 0.84 for inner state and low accuracy of 33% and F1-score of 0.33 for communication attempt. Confusion matrices are shown in Figure 4.

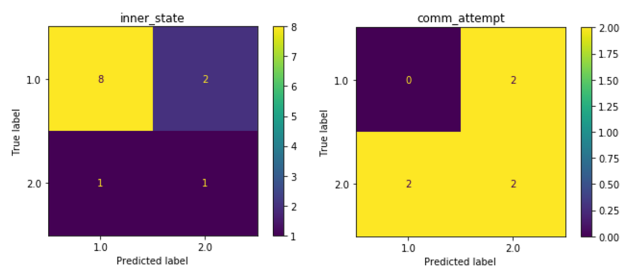


Figure 4: Confusion matrices for classifying rPPG instances using models trained on Empatica data. For inner state, the class values are 1.0="neutral" and 2.0="pleasure". For communication attempt 1.0="comment" and 2.0="demand".

6 CONCLUSION

We conducted an initial investigation of the connection between physiological signals and mental states of people with PIMD, attempting to classify their inner states and communication attempts. We used HRV features computed from the PPG obtained with an Empatica E4 wristband and investigated the performance of such models on instances obtained via rPPG. XGB has shown the best performance, achieving accuracy of 62% and F1 score of

0.59 for inner state, and accuracy of 48% and F1 score of 0.45 for communication attempt, notably surpassing the baseline majority classifier.

Limitations of our work lie in low number of instances for communication attempt and little variety in subjects, having just two for which annotations were available. Additionally, the evaluation using the rPPG is limited, as we had very few instances for which both high-quality segments of video and annotations were available. Thus, the focus of future work should be on gathering more data and conducting a more extensive evaluation of the methods, which is planned in the trial stage of the INSENSION project.

ACKNOWLEDGMENTS

This work is part of the INSENSION project that has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No. 780819. The authors also acknowledge the financial support from the Slovenian Research Agency (ARRS).

REFERENCES

- [1] Matej Cigale and Mitja Luštrek. 2019. Multiple knowledge categorising behavioural states and communication attempts in people with profound intellectual and multiple disabilities. In *Aml (Workshops/Posters)*, 46–54.
- [2] Matej Cigale, Mitja Luštrek, Matjaž Gams, Torsten Krämer, Meike Engelhardt, and Peter Zentel. 2018. The quest for understanding: helping people with PIMD to communicate with their caregivers. *INFORMATION SOCIETY-IS 2018*.
- [3] Torsten Krämer and Peter Zentel. 2020. Expression of emotions of people with profound intellectual and multiple disabilities. A single-case design including physiological data. *Psychoeducational Assessment, Intervention and Rehabilitation*, 2, 1, 15–29.
- [4] Richard D Lane, Kateri McRae, Eric M Reiman, Kewei Chen, Geoffrey L Ahern, and Julian F Thayer. 2009. Neural correlates of heart rate variability during emotion. *Neuroimage*, 44, 1, 213–222.
- [5] Junoš Lukan, Martin Gjoreski, Heidi Mauersberger, Annekatrin Hoppe, Ursula Hess, and Mitja Luštrek. 2018. Analysing physiology of interpersonal conflicts using a wrist device. In *European Conference on Ambient Intelligence*. Springer, 162–167.
- [6] Stanley Schachter. 1964. The interaction of cognitive and physiological determinants of emotional state. In *Advances in experimental social psychology*. Volume 1. Elsevier, 49–80.
- [7] Gašper Slapničar, Erik Dovgan, Pia Čuk, and Mitja Luštrek. 2019. Contact-free monitoring of physiological parameters in people with profound intellectual and multiple disabilities. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*.
- [8] Poznań Supercomputing and Networking Center. 2017. The INSENSION project. <https://www.insension.eu/>.
- [9] Pieter Vos, Paul De Cock, Vera Munde, Katja Petry, Wim Van Den Noortgate, and Bea Maes. 2012. The tell-tale: what do heart rate; skin temperature and skin conductance reveal about emotions of people with severe and profound intellectual disabilities? *Research in developmental disabilities*, 33, 4, 1117–1127.

Energy-Efficient Eating Detection Using a Wristband

Simon Stankoski
Department of Intelligent Systems
Jožef Stefan Institute
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
simon.stankoski@ijs.si

Mitja Luštrek
Department of Intelligent Systems
Jožef Stefan Institute
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
mitja.lustrek@ijs.si

ABSTRACT

Understanding people's dietary habits plays a crucial role in interventions that promote a healthy lifestyle. For this purpose, a multitude of studies explored automatic eating detection with various sensors. Despite progress over the years, most proposed approaches are not suitable for implementation on embedded devices. The purpose of this paper is to describe a method that uses a wristband configuration of sensors to continuously track wrist motion throughout the day and detect periods of eating automatically. The proposed method uses an energy-efficient approach for activation of a machine learning model, based on a specific trigger. The method was evaluated on data recorded from 10 subjects during free-living. The results showed a precision of 0.84 and a recall of 0.75. Additionally, our analysis shows that by using the trigger, the usage of the machine learning model can be reduced by 80%.

KEYWORDS

Eating detection, wristband, energy efficient, activity recognition

1 INTRODUCTION

Understanding people's dietary habits plays a crucial role in interventions that promote a healthy lifestyle. Obesity, which is a consequence of bad nutritional habits and excessive energy intake, can be a major cause of cardiovascular diseases, diabetes or hypertension. Latest statistics indicate that obesity prevalence has increased substantially over the last three decades [1]. More than 600 million adults (13% of the total adult population) were classified as obese in 2014 [2]. In addition, the prevalence of obesity is estimated to be 23% in the European Region by 2025. Also, in 2017, it was reported that poor diet has contributed to 11 million deaths worldwide. Monitoring eating habits of overweight people is an essential step towards improving nutritional habits and weight management.

Another group of people that require monitoring of their eating behavior are people with mild cognitive impairment and dementia. They often forget whether they have already eaten and, as a result, eat lunch or dinner multiple times a day or not at all. It might cause additional health problems. Proper treatment of these issues requires an objective estimation of the time the meal takes place, the duration of the meal, and what the individual eats.

Wristband devices and smartwatches are increasingly popular, mainly because people are accustomed to wearing watches, which makes the wrist placement one of the least intrusive body placements to wear a device. Additionally, the cost of these devices is relatively low, which makes them easily accessible to everyone. However, these devices offer limited computing power and battery life, which makes the implementation of a smart feature as eating detection on such a device a challenging task.

This paper describes a method for real-time eating detection using a wristband. The proposed method detects periods and duration of eating. The output from the method can be used to track frequency of eating and could serve to start methods for counting food intakes.

The work done in this study is important for the following reasons. We developed a trigger that can reduce the usage of the machine learning procedure, meaning that our method will not greatly affect the battery life of the device. Additionally, we evaluated different machine learning algorithms in terms of accuracy and model size. The method was evaluated on data recorded in real-life from 10 subjects.

2 RELATED WORK

Recent advancements in wearable sensing technology (e.g., commercial inertial sensors, fitness bands, and smartwatches) have allowed researchers and practitioners to utilize different types of wearable sensors to assess dietary intake and eating behavior in both laboratory and free-living conditions. A multitude of studies for the detection of eating periods have been proposed in the past decade. Mirtchou et al. [3] explored eating detection using several sensors and combining real-life and laboratory data. Edison et al. [4] proposed a method that recognizes intake gestures separately, and later clusters the intake gestures within 60-minute intervals. The method was evaluated on real-life data. Dong et al. [5] proposed a method for eating detection in real-life situations based on a novel idea

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
Information Society 2020, 5–9 October 2020, Ljubljana, Slovenia
© 2020 Copyright held by the owner/author(s).

that meals tend to be preceded and succeeded by periods of vigorous wrist motion. Amft et al. [6] presented an accurate method for eating and drinking detection using sensors attached to the wrist and upper arm on both hands. Navarathna et al. [7] combined sensor data from a smartwatch and a smartphone, which resulted in improved eating detection accuracy compared to only using smartwatch data. Kyritsis et al. [8] proposed a deep learning based method that recognizes bite segments, which are used for construction of eating periods.

The work presented in this paper is an extension of our previous work [9], and the main novelty is an energy efficient approach for real-time eating detection.

3 METHOD

The proposed eating detection method consists of two parts, namely: a threshold-based trigger, used for activation of an eating detection machine learning procedure, and a machine-learning method that predicts whether eating took place.

3.1 Energy-Efficient Trigger

The recent advancements in the technological development and accessibility of wearable devices bring new opportunities in the field of human activity recognition (HAR). However, the limited battery life and computational resources remain a challenge for real-life implementation of advanced HAR applications. Using a machine learning based model for eating detection that is working all the time results in a rapid battery drain. Therefore, we designed a threshold-based trigger that activates the machine learning model only when specific criteria are met. The main concept behind the trigger is to only select moments when the human is making a movement with his hand towards the head.

For this purpose, we used data from an accelerometer. This sensor provides information about the wristband's orientation from which we can see whether the hand is oriented towards the head. The recent accelerometers that are used in battery-limited devices can store acceleration values in their internal memory without interacting with the main chip of the microcontroller.

The first step of trigger implementation is to define the buffer size in the sensor's internal memory and the sensor's sampling frequency. Based on these two parameters, we enable the accelerometer to collect data for a specific time without interacting with the main chip of the microcontroller. This means that the main chip of the microcontroller could be in sleep mode for the predefined period. When the accelerometer's buffer is full, the accelerometer interrupts the main chip and transfers the stored acceleration data to it. We use the accelerometer's y-axis and z-axis to detect moments when the individual is moving the hand towards the face. Namely, we calculate the mean value for both axes, and if both of the values are above a predefined threshold value, the machine learning procedure for eating detection is activated. We used two axes for the trigger to reduce the possible situations in which our trigger is falsely activated. However, one can work only with one axis, which will result in more activated triggers. We could say that having more activated triggers is not desirable. However, if the eating detection method is not good enough to

detect eating after a trigger is activated during a meal, then the constraints of the trigger should be reduced.

The next step is the definition of stopping criteria for the machine learning model. The idea here is to stop the machine learning procedure after a specific number of windows if there is no eating detected. Each time our trigger is activated, the machine learning procedure is turned on for the next three buffers of data. The machine learning procedure is stopped if there is no positive prediction in any of the three windows. However, if there is at least one positive prediction, the machine learning procedure continues to work for another three new buffers. Also, the number of windows for which the machine learning procedure is active was experimentally obtained.

3.2 Machine-Learning Procedure

A detailed description of the used method can be seen in [9]. The method is based on machine learning and consists of the following steps: filtering the accelerometer and gyroscope data coming from the wristband, segmentation of the filtered data, feature extraction, feature selection, two stages of model training and predictions smoothing.

In the first step, the raw data were filtered with a 5th order median filter to reduce noise. Furthermore, the median filtered data was additionally filtered with low-pass and band-pass filters. Hence, we ended up with three different streams of data, median, low-pass and band-pass filtered data.

The accelerometer and gyroscope data were segmented using a sliding window of 15 seconds with a 3-second overlap between consecutive windows. This means that once we have 15 seconds of data, the buffer is adjusted to only store 3 seconds of new data. After that, each time the buffer is full, we add the new 3 seconds of data to the previous 15 seconds window and we drop the oldest 3 seconds from it. The reason for the length of the window is that it needs to contain an entire food intake gesture [10].

After the segmentation step, we extracted three different groups of features. Also, we included a feature selection step to improve the computational efficiency of the method, to remove the features that did not contribute to the accuracy and to reduce the odds of overfitting.

The training procedure for the method used in this study consists of three stages. The first two aim at training an eating-detection models on an appropriate amount of representative eating and non-eating data. The third step smooths the predictions of the model.

4 DATASET AND EXPERIMENTAL SETUP

For this study, we recorded data from 10 subjects (8 male and 2 female), ranging in age from 20 to 41 years. The data were recorded using a commercial smartwatch Mobvoi TicWatch S running WearOS, providing 3-axis accelerometer and 3-axis gyroscope data sampled at 100 Hz. The technical description of the sensors from the smartwatch shows that the recorded data is compatible with our target wristband for which we are developing our eating detection method. Additionally, the use of a commercially available smartwatch was an easier option for recording data. The collected dataset contains recordings

from usual daily activities performed by the subjects, including eating. The subjects were wearing the smartwatch on their dominant hand while recording. The smartwatch had an application installed on it, which enabled them to label the beginning and the end of each meal. There were no limitations about the type of meals the subjects could have while recording, which resulted in having 70 different meals included in the dataset. Furthermore, the subjects were also asked to act naturally while having their meals, meaning talking, gesticulating, using the smartphone, etc. The total data duration is 161 hours and 18 minutes, out of which 8 hours and 19 minutes correspond to eating activities.

For evaluation, the LOSO cross-validation technique was used. In other words, the models were trained on the whole dataset except for one subject on which we later tested the performance. The same procedure was repeated for each subject in the dataset. The results obtained using this evaluation technique are more reliable compared to approaches where the same subject's data is used for both training and testing, which show excessively optimistic results.

As mentioned before, smartwatches offer limited resources, one of which is the size of the RAM memory. Therefore, we analyzed models with different sizes to see whether the bigger and more complex models provide higher accuracy. We tested the performance of four different machine learning algorithms, Random Forest [11], Decision Tree [12], Logistic Regression [13] and LinearSVC [14].

We analyzed the following evaluation metrics: recall, precision and F1 score. These evaluation metrics are the most commonly used metrics for classification tasks like ours and give a realistic estimate of the efficacy of the algorithm. Also, the final results were obtained from the whole recordings by each subject. The reason for this is mainly to give a real picture of how good the developed method is in real-life settings.

5 RESULTS

The primary use of the trigger is to reduce the activity of the machine learning procedure. However, for the efficiency of the trigger, a very important requirement is when and how often the trigger is activated during a meal. In order to achieve accurate predictions, we want the trigger to be activated as soon as the meal is started. Additionally, the percentage of activated triggers during a meal should be bigger compared to noneating segments. For this purpose, we explored which window size works best with our trigger. Table 1 shows the results achieved in the conducted experiments. We tested two different window sizes with two slide values for each window, resulting in a total of four combinations.

Table 1: Different window size for the trigger procedure.

Window and slide size	Trigger activation time	% of activated triggers	Meals detected
3 - 1	36 s	34.2	68/70
3 - 3	41 s	32.6	68/70
15 - 3	48 s	42.0	55/70
15 - 5	41 s	42.0	54/70

Table 2: Results of eating detection procedure achieved with different algorithms and their model size.

Algorithm	Precision	Recall	F1 score	Model size
Random Forest	0.84	0.75	0.79	36339 KB
Logistic Regression	0.70	0.71	0.70	1.25 KB
LinearSVC	0.69	0.71	0.70	1.8 KB
Decision Tree	0.59	0.65	0.62	175 KB

The used combinations for the window and slide size are shown in the first column of the table. The second column shows the average time needed for the trigger to be activated for the first time after a meal is started. The third column shows the average percentage of triggered windows during a meal. These two columns were used as a metric for selecting the optimal size of a window and slide between the windows. The last column shows the number of meals when the trigger was activated. The values for the second and third columns were obtained only from the meals for which the trigger was activated. Row-wise comparison between these two columns shows the results obtained with each different combination of a window and slide. We can see that the most optimal combination regarding the average time needed for a trigger to be activated after a meal is started is a window size of 3 seconds with a slide of 1 second between two windows. Therefore, in our further analysis, we used this combination. The optimal window size of 3 seconds is expected if we have in mind that the usual intake gesture lasts around 2 seconds. Longer windows fail to detect the gesture while having a meal because usually we have two or three intakes in 15 seconds and the mean value over the whole window is low.

Table 2 shows the final results obtained using the whole method described in Section 2. Row-wise comparison between the used evaluation metrics shows the results obtained using the different algorithms shown in the first column. Additionally, the last column of the table represents the final model size. We can clearly see that the results achieved with Random Forest are better than the remaining algorithms. However, if we compare the model size of the best performing algorithm with the remaining algorithms we can say that the results achieved using Logistic Regression and LinearSVC are acceptable. Additionally, the precision value of 0.84 shows that the combination of trigger and machine learning procedure can differentiate between eating and noneating segments. However, the recall value of 0.75 suggests that a more accurate method regarding the eating periods is needed.

We also analyzed how much time each of the previously described algorithms was active during the noneating period. The results from this experiment are shown in Table 3. Additionally, in this table we can see the false positive rate during the noneating period. The best results are achieved using a Random Forest classifier, which is active only 20% of the whole noneating period. This means that our trigger-based procedure reduces the usage of the machine-learning procedure for 80%. However, this number also depends on the detection method because once it is activated, the eating predictions extend the active time of the method.

Table 3: Comparison of active time and false positive rate of the machine learning algorithms during noneating period.

Algorithm	Active time during noneating period	False positive rate
Random Forest	20%	1.36%
Logistic Regression	22%	2.18%
LinearSVC	22%	2.34%
Decision Tree	23%	3.93%

6 CONCLUSION AND FUTURE WORK

In this paper, we presented a method that can accurately detect eating moments using a 3-axis accelerometer and gyroscope sensor data. Our method consists of an energy-efficient trigger and a machine-learning procedure, which is started only after the trigger is activated. We evaluated this method using a dataset of 70 meals from 10 subjects. The results from the LOSO evaluation showed that we are able to recognize eating with a precision of 0.84 and recall of 0.75.

The presented results are important because both the training and the evaluation data were recorded in uncontrolled real-life conditions. We want to emphasize the real-life evaluation since it shows the robustness of the method while dealing with plenty of different activities that might be mistaken for eating as well as recognizing meals that were recorded in many different environments while using many different utensils. The proposed method can also deal with interruptions while having a meal, such as having a conversation, using the smartphone, etc. Additionally, we believe that the energy efficiency of the proposed method is very important. The proposed technique uses a trigger to activate the machine learning procedure and it is able to reduce the active time of the machine learning procedure for almost 80%. If we have in mind that the wristbands are devices with limited resources, we could say that even small reductions in resource usage can be significant for longer battery life.

The initial results achieved in this study are encouraging for further work in which we expect to improve the eating detection method. In the near future, we plan to optimize our machine learning procedure to detect eating periods more accurately once the trigger is activated. Furthermore, we want to overcome the problem with false positives predictions. For this problem, we believe that a more sophisticated method for selecting representative noneating data will help to recognize the problematic activities and directly include them in the training data. Also, we plan to investigate personalized threshold values. We believe that personalized values for the threshold will help

to activate the trigger during eating periods more easily. Additionally, this could reduce the activation of the machine-learning procedure during non-eating periods. Also, we plan to explore memory efficient methods for storing the models in memory.

ACKNOWLEDGMENTS

This work was supported by the WellCo and CoachMyLife projects. The WellCo project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 769765. The CoachMyLife project has received funding from the the AAL programme (AAL-2018-5-120-CP) and the Ministry of Public Administration of Slovenia.

REFERENCES

- [1] World Health Organization. World Health Statistics 2015. Luxembourg, WHO, 2015
- [2] Public Health England. Data Factsheet: Adult Obesity International Comparisons. London, 2016. http://webarchive.nationalarchives.gov.uk/20170110165728/http://www.noo.org.uk/NOO_pub/Key_data
- [3] M. Mirtchouk, D. Lustig, A. Smith, I. Ching, M. Zheng, and S. Kleinberg. Recognizing eating from body-worn sensors: Combining free-living and laboratory data. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(3):85:1–85:20, Sept. 2017
- [4] E. Thomaz, I. Essa, and G. D. Abowd. A practical approach for recognizing eating moments with wrist-mounted inertial sensing. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp '15*, pages 1029–1040, New York, NY, USA, 2015. ACM.
- [5] Y. Dong, J. L. Scisco, M. Wilson, E. Muth, and A. W. Hoover. Detecting periods of eating during free-living by tracking wrist motion. *IEEE Journal of Biomedical and Health Informatics*, 18:1253–1260, 2014
- [6] O. Amft, H. Junker, and G. Troster. Detection of eating and drinking arm gestures using inertial body-worn sensors. In *Ninth IEEE International Symposium on Wearable Computers (ISWC'05)*, pages 160–163. IEEE, 2005.
- [7] P. Navarathna, B. W. Bequette, and F. Cameron. "Wearable Device Based Activity Recognition and Prediction for Improved Feedforward Control." in *Proceedings of the American Control Conference*, 2018, doi: 10.23919/ACC.2018.8430775.
- [8] K. Kyritsis, C. Diou, and A. Delopoulos. "Detecting Meals in the Wild Using the Inertial Data of a Typical Smartwatch," in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, 2019, doi: 10.1109/EMBC.2019.8857275
- [9] Stankoski S, Resçiç N, Mezić G, Lustrek M. Real-time Eating Detection Using a Smartwatch. In *EWSN 2020 Feb 17* (pp. 247-252).
- [10] Xu Ye, Guanling Chen, and Yu Cao. Automatic eating detection using head-mount and wrist-worn accelerometers. In *2015 17th International Conference on E-health Networking, Application Services (HealthCom)*, pages 578–581, Oct 2015.
- [11] T. K. Ho. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE, 1995.
- [12] P. H. Swain and H. Hauska. "The decision tree classifier: Design and potential," in *IEEE Transactions on Geoscience Electronics*, vol. 15, no. 3, pp. 142-147, July 1977, doi: 10.1109/TGE.1977.6498972.
- [13] Lee, Youngjo, John A. Nelder, and Yudi Pawitan. *Generalized linear models with random effects: unified analysis via H-likelihood*. Vol. 153. CRC Press, 2018.
- [14] Chang, Chih-Chung, and Chih-Jen Lin. "LIBSVM: A library for support vector machines." *ACM transactions on intelligent systems and technology (TIST)* 2.3 (2011): 1-27.

Comparison of Methods for Topical Clustering of Online Multi-speaker Discourses

Vid Stropnik
University of Ljubljana,
Faculty of Computer and
Information Science,
Velenje, Slovenia
vs6309@student.uni-lj.si

Zoran Bosnić
University of Ljubljana,
Faculty of Computer and
Information Science,
Ljubljana, Slovenia
zoran.bosnic@fri.uni-lj.si

Evgeny Osipov
Luleå University of Technology,
Department of Computer Science,
Electrical and Space Engineering,
Luleå, Sweden
evgeny.osipov@ltu.se

ABSTRACT

Discussions held on online forums differ from traditional text documents in several ways. In addition to individual text-bodies (submission comments, forum posts etc.) being very short, they also have multiple messengers, each of whom may exhibit unique patterns of speech. Consequently, state of the art methods for text summarization are often rendered inapplicable for these sorts of corpora. This paper evaluates the topic-clustering algorithm used in the state-of-the-art online comment clustering techniques, as parts of commonly used summarizer models. It proposes two alternative, vector-based approaches and presents results of a comparative external analysis, concluding in the three methods being comparable.

KEYWORDS

latent Dirichlet allocation, word embeddings, GloVe, hyperdimensional computing, self-organized maps, topical clustering, clustering evaluation, discussion summarization

1 INTRODUCTION

User generated comments carry a great amount of useful information. Big data researchers have successfully used them to predict stock market volatility [1] and predict the characteristics of such comments that perform the best on a given online platform [2]. User comments can also offer vast amounts of complementary information, as well as being forms of information surveillance, entertainment or social utility [3]. Existing mechanisms for displaying comments on websites do not scale well and often lead to *cyberpolarization* [4]. Furthermore, they are platform-specific and often fail to offer an overall image of the topics discussed in a given comments section.

A comprehensive, easily understandable automatic summary of the online discourse at hand can be instinctively understood as a solution to this problem. This, however, is no easy task, seeing as these corpora are often very short and come from multiple

speakers. Consequently, traditional summarization methods do not translate well to these sorts of text bodies.

In Section 2 of this paper, the related work establishes the general framework that other authors generally use for the task at hand. It establishes the Latent Dirichlet Allocation (LDA) topic modeling algorithm as the current leading method for topical grouping of individual comments. These topical groups play a pivotal role in later summarization steps, also presented in Section 2.

In this paper, we externally evaluate and compare LDA versus two frameworks, using word representations in semantic vector space. We describe the analyzed methods in Sections 3 and 4. In Section 5, we describe the comparative evaluation methodology used to determine the applicability of each modeling technique and present our results. We follow it up by discussing further work in the conclusion of this paper.

2 RELATED WORK

Online discussion summarization is a field that has not been addressed directly by many authors. One group of works [5-7] have roughly described a three-step process, commonly presented as the state of the art. The approaches includes a topical clustering of all the observed comments, establishing a ranking method for determining the most salient ones in each cluster, and later summarizing this selection. Between them, the authors confidently establish Latent Dirichlet Allocation (LDA) topic modeling as the most human like grouping algorithm. Further work also proposes a novel graph-based linear regression model based on the Markov Cluster Algorithm (MCL), [8] which outperforms LDA, but uses the knowledge of multidomain knowledge bases for implementation. While we argue that extractive summarization is not an ideal method for the analysis of multi-speaker corpora, the first step of identifying and topically clustering individual comments in each comment section is assumed as a required step towards successful summarization of the topics discussed therein.

To the best of our knowledge, popular NLP word embedding algorithms (i.e. *word2vec*, *GloVe*) have not been used directly for comment summarization applications up until now. Similarly, neither have hyperdimensional representations, another topic of interest.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
Information Society 2020, 5–9 October 2020, Ljubljana, Slovenia
© 2020 Copyright held by the owner/author(s).

3 NLP METHODS

In this work, we examine three distinct topical clustering models, the output of which is always a set of comment clusters, given a multi-comment input.

The first is an LDA model, using a Term frequency – inverse document frequency (TFIDF) word representation as an input. In this representation, the comments were hard-clustered into the groups, determined by the degree of membership of which had been the highest in a soft-clustering approach, provided by the LDA model.

The second examined model uses *GloVe* word embeddings clustered with the k-means clustering algorithm, thus portraying words in semantic vector space using information of contexts in which words often appear.

The third model creates Hyperdimensional representations of words, mapped them into a two-dimensional topology using the self-organized maps algorithm and then clustered it like the preceding model. This approach is the least explored for this use-case and is inspired by the observed differences between the functionality of the human brain and the traditional von Neumann architecture for modern computing.

We performed the comparative evaluation of the models on the *Reddit Corpus (by subreddit)* dataset, provided by the Cornell Conversational Analysis Toolkit (Convokit)¹. Five *Conversations*, corresponding to as many threads on the website Reddit were extracted from the corpus. We selected threads, discussing topics from different subject domains, where each contained at least 50 non-removed comment text bodies. Two human annotators were then asked to manually identify topical clusters in the selected *Conversations*. The comment texts were provided to them in the form of a set of numbered text files, containing only the text data in chronological order of submission. Reddit post titles or other metadata were not available to the annotators and no guidance was given as to the number of topics required. The clusterings were examined as-is, with no singleton removal performed.

We describe the NLP techniques used to create the three clustering models in the following subsections, with external evaluation results being presented in Section 5.

3.1 Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) is a topic modeling technique initially proposed in the context of population genetics, but later applied in machine learning in the early 21st century. It assumes a generative process of documents as random mixtures over a collection of latent topics. Each of these topics, in turn, is characterized by a certain distribution over words. A topic model can be created by estimating the per document distribution of topics θ and the per topic distribution over words ϕ . [9] Many methods, such as variational inference, Bayesian parameter estimation [9] and Collapsed Gibbs sampling [10], have been used to approximate these values. In the end, they all boil down to maximizing the model's probability of creating the exact documents, provided to it in the input, assuming the knowledge of the number of topic distributions.

3.2 Word Embeddings

Word Embeddings is a collective name for a set of language modeling and feature learning techniques, yielding word representations using vectors, the relative similarities of which correlate with the semantic similarity of the represented words. These meanings are extracted from the contexts – fixed-size windows of preceding and succeeding words, in which individual words appear in in the training corpus. The generation of these vectors is achieved by Context counting [11] or Context prediction [12]. While there have been several claims of one of the methods for synthesizing word embeddings being superior over another, recent work implies the correspondence between these model-types [13]. Whichever way these word-vectors are created, they represent semantic meaning in vector space. Using algebraic similarity measures (in our case, cosine distance) on comment-word averages, the relative likeness of the examined comments' meanings is calculated. Comment clusters can then be created by clustering the semantic-space points into groups with high intra-cluster and low inter-cluster similarity. These groups represent topical clusters, used in our examination.

3.3 Hyperdimensional Computing

Hyperdimensional computing is a family of biologically inspired methods for representing and manipulating concepts and their meanings in high-dimensional space. Random Bipolar vectors of high, but fixed dimensionality (≥ 1000) are initialized as individual word representations and are then transformed in ways that represent semantically similar comments closer in the high-dimensional vector space, while the similarity of dissimilar comments is likely close to zero due to their inherent orthogonality. The methods used to transform these vectors are binding, bundling and permuting [14]. By using these methods, individual hyperdimensional vectors are created for each comment, encoding the used words and their position in the comment in the vector.

Similar to the clustering of word embeddings, semantically similar comment groups can be found by clustering, thus determining the outputs of the third model. However, the performance of this method did not yield comparative results at first. We hypothesised that this might be due to the high component count of the used vectors (more than double the dimensions of the Word Embedding approach), so a method of dimensionality reduction was examined, aiming to improve its results. It is described in the next sub-section.

3.4 Self-Organized Maps

Self-organizing maps (SOM), also known as Kohonen networks are computational methods for the visualization and analysis of high-dimensional data. The output of the algorithm is a set of nodes, arranged in a certain topology that represents the nodes' mutual relation, with each node being represented with a weight vector of t dimensional components, with t corresponding to the uniform dimensionality of data being reduced [15]. As data representations in high-dimensional vector spaces are inherently vulnerable to sparseness, clustering outputs can differ in cases where the clustered data is first dimensionally reduced. Thus, we used the SOM algorithm to examine if the results (of the

¹ <https://convokit.cornell.edu/documentation/index.html/>

examination in Section 5) of any of the proposed frameworks can be improved by dimensionally reducing the vector representations prior to clustering.

SOM proved to drastically improve the performance of the Hyperdimensional computing model, while making the Word Embeddings-based model perform worse. Consequently, we only use SOM prior to clustering the HD-based approach in the evaluation, presented in Section 5.

4 IMPLEMENTATION

All implementational work was done with the *Python* programming language. All text corpora were pre-processed using the WordNetLemmatizer and PorterStemmer from NLTK.² Stop word removal was done in the pre-processing step using the topic modeling package Gensim³, which also provided the submodules for TFIDF and LdaModel, used for the implementation of Latent Dirichlet Allocation. GloVe word embeddings were provided as part of the NLP open-source library SpaCy⁴ as part of the “*en_core_web_md*” pretrained statistical model for the English Language. The SOM algorithm was implemented using the SimpSOM package⁵, with k-means clustering being provided by Scikit-Learn.⁶

5 EVALUATION

To analyze the applicability of LDA, Word Embeddings and dimensionally reduced Hyperdimensional computing for the discussed use-case, topical clustering outputs were created for 5 Reddit *Conversations*. Two human annotators also manually created topical groups for these conversations. The goal of our evaluation was to see which model created the most *human-like* clusters; consequently having the highest average agreement measure with the clustering samples, provided by the two annotators.

Topical clusters, created by the three models, were externally evaluated using four symmetric agreement measures: The V-Measure [16], The Fowlkes-Mallows Index [17], the Rand Index [18] and the Mutual information score [19]. The latter two were also adjusted for variance. For each examined model, the best performing number of topic clusters was selected. The agreement of the clustering output of each model was measured against both of the manual clusterings, with the *per annotator* average of each metric being the final output.

Figure 1 shows the result scores of all four metrics for each analyzed method. In the top row, the average agreement between the two annotators is also shown. This is, expectedly, higher than the average agreement between any examined model and the human outputs. A few takeaways can be addressed, examining the figure. Firstly, the different methods were successful to a varying degree, depending on the used metric, with each performing the best according to at least one. Secondly, when comparing their average relative success in relation to the agreement scores between Annotator A and Annotator B, we can see that their performances are very similar. This can be seen

even clearer in Figure 2, which shows each model’s performance with respect to the agreement score between the two human annotators. The percentage is calculated as an averaged sum of all four metric scores, weighted by the sum of these scores, achieved by the human versus human evaluation. In the figure, Word Embeddings can be seen as the best-performing approach, reaching 54.18 % of the Human agreement. The performance of LDA presented in Figure 2 is also comparable to that found in [5].

However, the difference in results between the best and the

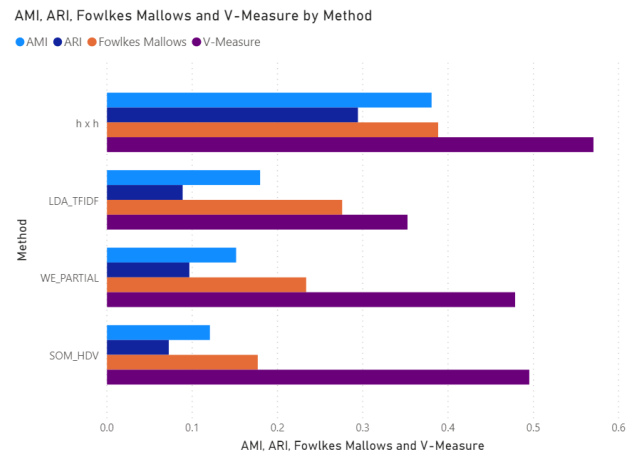


Figure 1: Visualization of agreement metric results between the human annotators (top) and the average annotator vs. model agreement (bottom three)

worst performing models being less than 7% of the total human agreement score, this metric is not enough to establish Word Embeddings as superior to LDA or indeed, dimensionally reduced High-dimensional computing. We can conclude that both Hyperdimensional computing and Word Embeddings can produce topical clusters, comparable to the current state of the art LDA method.

Semantic document representations performing as well as the state-of-the-art topic modeling framework using LDA opens up plentiful possibilities in the field of multi-speaker conversation analysis. Whereas topic modeling’s more direct approach of inferring latent conversation topics might be useful in their discovery, the possibility of applying algebraic functions to individual comment vectors might enable further topic mining and experimentation. While the k-means clustering algorithm requires a desired number of clusters at input, similar to LDA, its job is not to encode semantics in the Word Embedding or SOM-HDC framework. This means that an alternative clustering algorithm – one without the need for an input number of medoids - could be used for the task of grouping comments. This, in turn, would result in a truly unsupervised topical clustering framework. A comparative evaluation of these approaches is a field of interest in the future, as our non-conclusive experiments have

² <https://www.nltk.org/>

³ <https://radimrehurek.com/gensim/>

⁴ <https://spacy.io/>

⁵ <https://github.com/fcomitani/SimpSOM/>

⁶ <https://scikit-learn.org/>

already shown a vast variance in results when using different clustering approaches.

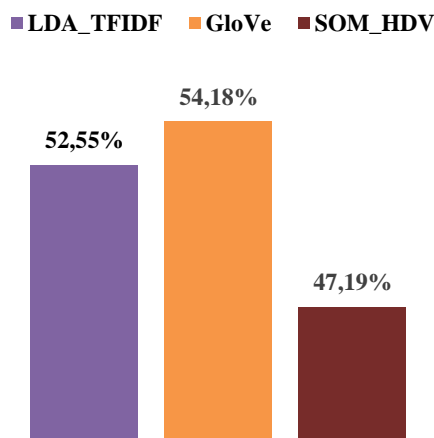


Figure 2: Percentage of the Human versus human agreement score achieved by each model (averaged between 4 agreement metrics)

6 CONCLUSION

In this article, we work from our hypothesis that popular semantics-laden vector representations of text data can be applicable in the established framework for extractive online discussion summarization. We present two models using different vector-based representation techniques and conclude that they are both comparable to the Latent Dirichlet Allocation topic modelling technique, used in most literature, with the Word Embeddings-based framework outperforming it in our external evaluations.

As mentioned in Section 2, the authors of this article argue that extractive summarizations are intrinsically less suitable when working with multi-speaker corpora. Our future work in this field includes the modeling of an abstractive summarizer framework, using the findings presented in this paper. Our intent is to use them in conjunction with graph-based approaches that take advantage of multidomain knowledge bases like DBpedia for both clustering and topic-labelling [8, 20].

Whether used in extractive or abstractive applications, we presume that the field will greatly benefit from our findings, seeing that the two vector-based representation frameworks open a plethora of new possibilities for other researchers. These include the detailed data manipulation using algebraic operations on individual comment vectors, as well as said vectors being suitable inputs for deep learning models using neural networks.

REFERENCES

- [1] W. Antweiler and M. Z. Frank, 'Is All That Talk Just Noise? The Information Content of Internet Stock Message Boards', *J. Finance*, vol. 59, no. 3, pp. 1259–1294, Jun. 2004, doi: 10.1111/j.1540-6261.2004.00662.x.
- [2] T. Weninger, 'An exploration of submissions and discussions in social news: mining collective intelligence of Reddit', *Soc. Netw. Anal. Min.*, vol. 4, no. 1, p. 173, Dec. 2014, doi: 10.1007/s13278-014-0173-9.
- [3] E. Go, K. H. You, E. Jung, and H. Shim, 'Why do we use different types of websites and assign them different levels of credibility? Structural relations among users' motives, types of websites, information credibility,

- and trust in the press', *Comput. Hum. Behav.*, vol. 54, pp. 231–239, Jan. 2016, doi: 10.1016/j.chb.2015.07.046.
- [4] S. Faridani, E. Bitton, K. Ryokai, and K. Goldberg, 'Opinion space: a scalable tool for browsing online comments', in *Proceedings of the 28th international conference on Human factors in computing systems - CHI '10*, Atlanta, Georgia, USA, 2010, p. 1175, doi: 10.1145/1753326.1753502.
- [5] C. Llewellyn, C. Grover, and J. Oberlander, 'Summarizing Newspaper Comments', *Proc. Eighth Int. AAAI Conf. Weblogs Soc. Media*, pp. 599–602, Jun. 2014.
- [6] Z. Ma, A. Sun, Q. Yuan, and G. Cong, 'Topic-driven reader comments summarization', in *Proceedings of the 21st ACM international conference on Information and knowledge management - CIKM '12*, Maui, Hawaii, USA, 2012, p. 265, doi: 10.1145/2396761.2396798.
- [7] E. Khabiri, J. Caverlee, and C.-F. Hsu, 'Summarizing User-Contributed Comments', presented at the International AAAI Conference on Weblogs and Social Media, pp. 534–537, Barcelona, Spain, Jul. 2011.
- [8] A. Aker *et al.*, 'A Graph-Based Approach to Topic Clustering for Online Comments to News', in *Advances in Information Retrieval*, vol. 9626, N. Ferro, F. Crestani, M.-F. Moens, J. Mothe, F. Silvestri, G. M. Di Nunzio, C. Hauff, and G. Silvello, Eds. Cham: Springer International Publishing, 2016, pp. 15–29.
- [9] D. Blei, A. Y. Ng, and M. I. Jordan, 'Latent Dirichlet Allocation', *J. Mach. Learn. Res.*, vol. 3, no. 4–5, pp. 993–1022, 2000, doi: 10.1162/jmlr.2003.3.4-5.993.
- [10] W. M. Darling, 'A Theoretical and Practical Implementation Tutorial on Topic Modeling and Gibbs Sampling', *Proc. 49th Annu. Meet. Assoc. Comput. Linguist. Hum. Lang. Technol.*, pp. 642–647, Dec. 2011.
- [11] J. Pennington, R. Socher, and C. Manning, 'Glove: Global Vectors for Word Representation', in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, 2014, pp. 1532–1543, doi: 10.3115/v1/D14-1162.
- [12] T. Mikolov, K. Chen, G. Corrado, and J. Dean, 'Efficient Estimation of Word Representations in Vector Space', *ArXiv13013781 Cs*, Sep. 2013, Accessed: Aug. 19, 2020. [Online]. Available: <http://arxiv.org/abs/1301.3781>.
- [13] A. Österlund, D. Ödling, and M. Sahlgrén, 'Factorization of Latent Variables in Distributional Semantic Models', in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, 2015, pp. 227–231, doi: 10.18653/v1/D15-1024.
- [14] D. Kleyko, E. Osipov, D. De Silva, and U. Wiklund, 'Distributed Representation of n-gram Statistics for Boosting Self-organizing Maps with Hyperdimensional Computing', in *Perspectives of System Informatics, 12th International Andrei P. Ershov Informatics Conference, Revised Selected Papers*, pp. 64–79, Novosibirsk, Russia, 2019.
- [15] T. Kohonen, T. S. Huang, and M. R. Schroeder, *Self-Organizing Maps*. Berlin, Heidelberg: Springer Berlin / Heidelberg, 2012.
- [16] A. Rosenberg and J. Hirschberg, 'V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure', in *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, Prague, Czech Republic, Jun. 2007, pp. 410–420, Accessed: Aug. 20, 2020. [Online]. Available: <https://www.aclweb.org/anthology/D07-1043>.
- [17] E. B. Fowlkes and C. L. Mallows, 'A Method for Comparing Two Hierarchical Clusterings', *J. Am. Stat. Assoc.*, vol. 78, no. 383, pp. 553–569, Sep. 1983, doi: 10.1080/01621459.1983.10478008.
- [18] W. M. Rand, 'Objective Criteria for the Evaluation of Clustering Methods', *J. Am. Stat. Assoc.*, vol. 66, no. 336, pp. 846–850, Dec. 1971, doi: 10.1080/01621459.1971.10482356.
- [19] N. X. Vinh, J. Epps, and J. Bailey, 'Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance', *J. Mach. Learn. Res.*, vol. 11, pp. 2837–2854, Oct. 2010.
- [20] I. Hulpus, C. Hayes, M. Karnstedt, and D. Greene, 'Unsupervised graph-based topic labelling using dbpedia', in *Proceedings of the sixth ACM international conference on Web search and data mining - WSDM '13*, Rome, Italy, 2013, p. 465, doi: 10.1145/2433396.2433454.

Machine Learning of Surrogate Models with an Application to Sentinel 5P

Michał Artur Szlupowicz
m.szlupowicz@gmail.com
Warsaw University of Technology,
Faculty of Physic
Warsaw, Poland

Jure Brence
jure.brence@ijs.si
Jožef Stefan Institute
Ljubljana, Slovenia

Jennifer Adams
jennifer.adams@esa.it
Φ-lab, ESA/ESRIN
Frascati, Italy

Edward Malina
edward.malina.13@alumni.ucl.ac.uk
Earth and Mission Science Division
ESA/ESTEC
Noordwijk, the Netherlands

Sašo Džeroski
saso.dzeroski@ijs.si
Jožef Stefan Institute
Ljubljana, Slovenia

ABSTRACT

Surrogate models are efficient approximations of computationally expensive simulations or models. In this paper, we report improvements of a framework for learning surrogates on input and output spaces with reduced dimensionality. We present non-linear embeddings and feature importance as additional methods for dimensional analysis and reduction. The choice of models for prediction is extended with two types of ensembles of decision trees. The performance of the additions is evaluated and compared with the original approaches on a dataset, generated by RemoTeC, a complex radiative transfer model.

KEYWORDS

spectral data, neural network, ensemble, surrogate model, dimensionality reduction

1 INTRODUCTION

The TROPOspheric Monitoring Instrument (TROPOMI) is an on board satellite instrument on the Copernicus Sentinel-5 Precursor satellite [9]. Its main objective is to provide accurate observations of atmospheric parameters, as the concentrations of atmospheric constituents. Those can be used to obtain better air quality forecasts and to monitor global trends. However, the retrieval of interesting attributes involves running a retrieval algorithm, such as RemoTeC [2, 8], based on “optimal estimation methods” that tend to be computationally very expensive [7].

Machine learning techniques can be used to learn surrogate models that approximate the outputs of intensive simulations and are much faster at making predictions [13]. A framework for learning surrogates of radiative transfer models has been developed [1]. Due to the high dimensionality of both input and output spaces, the framework employs dimensionality reduction - methods that find low-dimensional projections (embeddings) of data that preserve as much information as possible [4]. Predictive models are learned on input and output spaces with reduced dimensionality.

Despite promising results, the existing framework for learning surrogates is limited to simple feed-forward neural networks for

the task of prediction, while offering a choice between PCA and autoencoders to reduce dimensionality [4, 6, 3]. In this paper we present an extension of the framework with two types of ensembles of decision trees for prediction [4], as well as an evaluation of the performance and utility of three additional algorithms for dimensionality analysis and dimensionality reduction: t-SNE [11], UMAP [12] and feature importance based on random forests [10].

2 DATASET

The training dataset was generated using the RemoTeC tool and in total consists of 50000 samples. Each input state vector contains a set of atmospheric parameters: solar zenith angle (SZA), albedo, temperature, pressure, aerosols and profiles of the CH₄, CO and H₂O gases (in total 125 dimensions). The sampling of the data ensures that the data covers the entire range of conditions that S5P/TROPOMI is expected to encounter. Exploratory data analysis reveals three dimensions with zero variance. Removing them results in a dataset with a 122-dimensional input space.

The output training data was created using the RemoTeC RTM in the S5P/TROPOMI Shortwave InfraRed (SWIR3) band. Each target vector consists of an infrared spectrum with 834 dimensions.

3 SURROGATE MODELS

The framework for learning surrogates is capable of learning both forward and backwards models. The former predict spectra, given atmospheric parameters. The latter reverse this process and learn to approximate atmospheric parameters that produce a given spectrum, which is useful for optimizing parameters of the RemoTeC simulation. Surrogates are generally predictive models that map directly between input and output data of a simulation or computationally expensive model. They offer much faster predictions at the cost incurring a prediction error. However, when the data is high dimensional and contains many samples, the computational cost of training and prediction can still be non-trivial. In such cases, methods of dimensionality reduction can offer not only time savings, but also improvements in predictive performance. In our framework, we employ dimensionality reduction to atmospheric parameters, as well as to the spectral space. Predictive models learn to map between reduced spaces. An inverse transformation is performed on predictions in the reduced space to obtain predictions in the original output space. For that reason, dimensionality reduction algorithms must provide

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

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

© 2020 Copyright held by the owner/author(s).

an inverse transformation in order to be useful as a component of a surrogate model in our framework.

3.1 Dimensionality Reduction

A high number of dimensions makes a problem much harder for many machine learning algorithms due to the curse of dimensionality. For this reason, we have tried a range of dimensionality reduction (DR) methods on our data before performing training on them. DR methods are (potentially unsupervised) algorithms that try to find a projection of the data to a lower dimension of space that preserve as much information as possible.

A lower number of dimensions helps reduce computation time and often even improves the predictive performance of models. Furthermore, DR methods can also be used to visualize high dimensional data by finding an informative projection into two dimensions that is understandable to humans. Some algorithms, such as t-SNE or UMAP, serve especially this purpose.

Principal Component Analysis (PCA) is one of the most popular dimensionality reduction methods [4]. PCA finds linear projections to a lower-dimensional subspace so that variance in the data is maximized. Visualizing the ratio of variance, covered by individual principal components is a way of assessing the intrinsic dimensionality of the data, as shown in Figure 1. We see that, for the 122-dimensional atmospheric parameter space, we need:

- 23 dimensions to explain 95% of the variance,
- 45 dimensions to explain 99% of the variance,
- 73 dimensions to explain 99.9% of the variance,

and for the output 834-dimensional spectral space:

- 1 dimension to explain 95% of the variance,
- 2 dimensions to explain 99% of the variance,
- 9 dimensions to explain 99.9% of the variance.

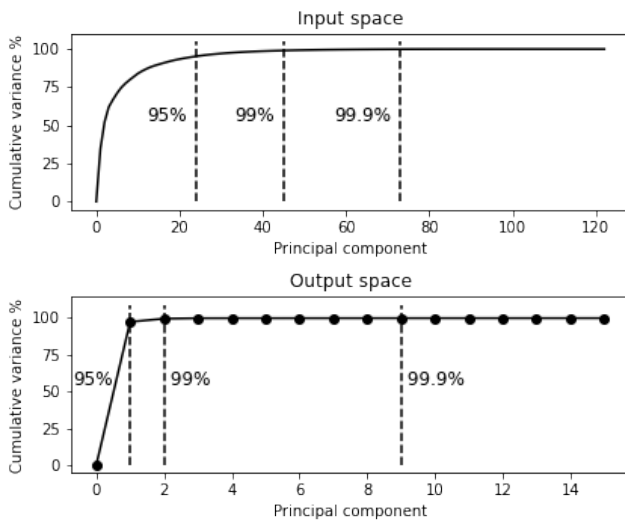


Figure 1: Dependence of the cumulative relative variance on the number of principal components for both the input and the output space.

Autoencoders (AE) [3] are a type of artificial neural network used to learn low dimensional representations. AE are trained to reproduce input data on the output of the network after passing through a bottleneck in the network architecture. To prevent autoencoders from memorizing the training dataset, a variety

of regularization techniques can be employed. One of options is adding artificial noise to the input data, which forces the network to generalize.

In our framework, we employ this kind of autoencoder, often referred to as a denoising autoencoder, by adding Gaussian noise with mean 0 and standard deviation 0.1 to input data during the training process. A more thorough investigation of the effect of this technique on the predictive power can be found in [1]. For both atmospheric parameters and the spectral space, we used the same 7 layers architecture with an appropriate size of input and output layers. The architecture can be summarized as:

- input layer of size N_0 + Gaussian noise
- dense layer of size $N_1 < N_0$ and ReLu activation
- dense layer of size $N_2 = \frac{1}{2}N_1$ and ReLu activation
- dense embedding layer of size N_3 and linear activation
- dense layer of size N_2 and ReLu activation
- dense layer of size N_1 and ReLu activation
- output layer of size N_0 and linear activation

The t-Distributed Stochastic Neighbor Embedding (t-SNE) [11] is a non-linear unsupervised technique for high dimension data visualization that can model complex, non-linear dependencies. t-SNE places points that are similar in the original space close together in the embedding layer with a high probability, while placing dissimilar points close together with only a low probability. Since t-SNE is a stochastic and non-parametric method there is no way to perform a reverse transformation from the embedding space to the original space. This excludes the method from use as part of the surrogate modelling process. It can, however, be useful for visualizing the dataset. Another disadvantage of t-SNE is its high computational complexity.

Uniform Manifold Approximation and Projection (UMAP) [12] is another dimension reduction technique used for dataset visualizations, constructed from a theoretical framework based in Riemannian geometry and algebraic topology. UMAP preforms similarly to t-SNE, but preserves more of the global data structure with superior run time performance. As is the case with t-SNE, UMAP does not allow for reverse transformations, which means we can not use it to learn surrogates. However, visualizations using UMAP allowed us to gain useful insights into the structure of our dataset.

3.2 Prediction Models

One of the predictors we used in our experiment was a feed-forward neural network (NN). We have chosen an architecture, consisting of 2 hidden full connected layers with ReLu activation functions and linear activation on the output layer [6].

Random Forest (RF) is an ensemble learning technique suited for both regression and classification problems. It uses sample bagging and feature sampling methods to train a set of decision trees. Prediction is performed by averaging over predictions from the individual regression trees. The main advantage of RF over a simple decision tree is the much better generalization. We decided to use this kind of predictor, because it is capable of performing multi target regression [10].

Extra Random Trees (ET) is a technique very similar to random forests, with two main differences. First, it uses the whole dataset for training individual trees instead of using bags of samples. Second, it uses random cuts for each split, instead of using the optimal one (in case of Gini or Entropy reduction). It has been shown to perform better than random forests for some problems [5].

4 EXPERIMENT

Our experiment is composed of three parts. In the first two, we employ methods of dimensionality reduction as a way to gain insight and understanding about our dataset and problem. The third part is an empirical evaluation of different combinations of methods for dimensionality reduction and prediction, aiming to identify the one that offers the best predictive performance on unseen data.

4.1 Visualization

We applied the UMAP and t-SNE visualization techniques to both atmospheric parameters and spectrum data. As expected, both methods showed clusters in the atmospheric parameters data. In the spectrum data space, UMAP identified a structure in the data, depicted in figure: 2. A comparison of the data points sampled from different clusters shows a large difference in the scale of individual data points. This is likely one of the reasons why such a high variance is concentrated in the first principal component (as seen in Figure 1).

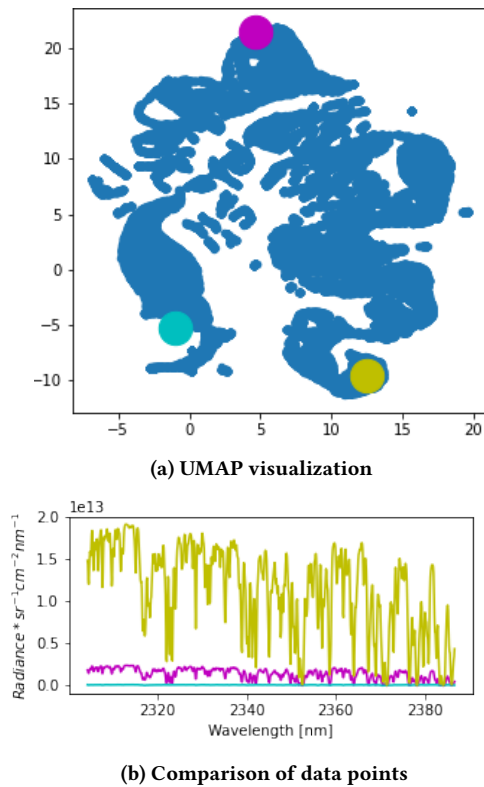


Figure 2: UMAP visualization of the spectrum data.

4.2 Feature Importance

The main advantage of using tree-based models over neural networks is their interpretability. While the ability to be understood by a human is lost when moving to an ensemble from a single tree, random forests can be very useful for estimating the importance of individual features for prediction. We trained a random forest predictor on the full dataset and visualized feature importance values in Figure 3. We see that 70% of feature importance is accumulated in just two dimensions. This corresponds well to the PCA estimate of most variance being encompassed by

two principal components. Only about half of the features are assigned non-negligible importance. The features identified by this approach warrant further investigation by domain experts.

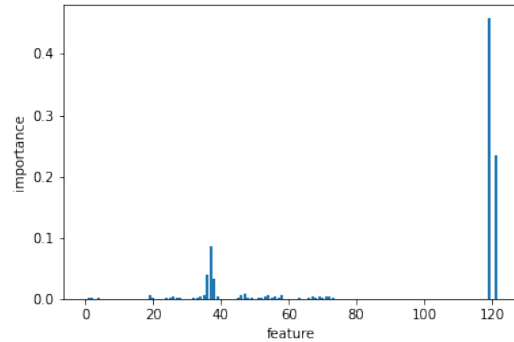


Figure 3: Random forest predictor importance of atmospheric data features.

4.3 Regression

To compare different regressors and methods of dimensionality reduction, we performed forward and backward predictions using neural network, random forest and extra random trees for both autoencoder and PCA embeddings. We reduced the dimensionality of the input space from 123 to 73 and the dimensionality of the output space from 834 to 9. These values correspond to 99.9% explained variance when using PCA. The noise level of the autoencoder was set to $\sigma = 0.1$. A more thorough study of the effects of these parameters can be found in [1]. We compare the predictive power of various combinations of either AE or PCA for dimension reduction, and either neural network, random forest or extra trees as a predictive model, using 10-fold cross validation. In Table 1 we compare the results, using the coefficient of determination as the evaluation metric [4]:

$$R^2 = 1 - \frac{MSE(\text{model})}{\text{variance}(\text{training set})}$$

Table 1: Coefficient of determination for various combinations of dimensionality reduction methods (DR) and predictive models (PM), estimated by 10-fold cross validation.

PM / DR	forward		backward	
	AE	PCA	AE	PCA
NN	0.9995	0.9998	0.8454	0.9206
RF	0.8931	0.9937	0.9267	0.9311
ET	0.9228	0.9958	0.9370	0.9510

For the forward model, the best performance of $R^2 = 0.9998$ is achieved by a neural network, mapping between spaces reduced by PCA. For the backward model, the best performing model are extra trees, paired with PCA, achieving $R^2 = 0.9510$. Both represent very satisfactory and promising models to employ as surrogates for radiative transfer modeling. From Table 1, we can also see that PCA outperformed autoencoders in all cases, while also being much faster to compute. The comparison of predictive models is not as simple. For the forward model, the neural

network is the best, but only by a small margin. For the backward model, the differences are larger, with the neural network performing the worst. The performance of random forests was between the performances of the other two predictive models for both the forward and the backward problem.

Since one of the main uses for surrogate models is speeding up computation, time complexity is an important consideration. The main disadvantage of neural networks is the computational complexity required for both training and prediction. An autoencoder takes about ten times as long to transform a data point to the embedding space than PCA. For predictive models, the neural network used in this study needed approximately three times as long to make a prediction than random forests and extra trees, which had a similar time complexity. Nonetheless, making predictions for a test set of 5000 points using any of the described surrogates takes up to one second, while running the full RemoTeC simulation requires several hours of computation.

When comparing with the evaluation results reported for the original framework in [1], the performances in this paper are slightly worse. The reason is the fact that the original study reduced the dimensions of the input space to 102 and the output space to 50 dimensions. In this study we focused on further reducing the dimensions and reduced the dimension of the input space to 73 dimensions and the output space to 9 dimensions. It is an interesting observation that for different dimensionalities, the best performance is achieved by different algorithms.

5 DISCUSSION AND FURTHER WORK

The original framework for learning surrogates on input and output spaces with reduced dimensionality showed high predictive and computational performance on the RemoTeC dataset. The results were very promising for applications in data analysis for Earth Observation missions as a way to dramatically speed up computation without sacrificing much accuracy. However, no single model and approach is the best for every dataset and application, which made the limited scope of options in the original framework a potential downside. With the work presented in this paper, the range of methods available has been extended. Since the choice of algorithms for dimensionality reduction on the input and output spaces, as well as the choice of prediction model for both the forward and the backward model are all independent from each other, the number of combinations of algorithms available is considerable. Furthermore, the dimension analysis enabled by UMAP, t-SNE and feature importance represents a new way of assessing intrinsic dimensionality and making a more informed choice of the number of target dimensions.

The paper presents an evaluation of the performance of various included methods on the RemoTeC dataset. However, each of the analyzed algorithms is defined by a number of hyperparameters, which is especially true for neural networks and autoencoders. Furthermore, the dimensions of the reduced input and output spaces can also be considered hyperparameters of the framework. For the presented evaluation we chose the hyperparameters based on values reported in previous work and to some degree optimized them manually. A more rigorous study is required that employs automated hyperparameter optimization in order to compare the available algorithms fairly and arrive at a reliable conclusion of what is the best approach to modeling the RemoTeC simulation.

Finally, in this study we touched upon the subject of estimating feature importance using random forests in order to gain

insight about the data. However, feature importance can also be used to compute feature rankings and perform feature selection, which can be considered as another method of dimensionality reduction. In further work, it might be worthwhile to investigate this approach further and include it as an option in the framework for learning surrogates.

6 ACKNOWLEDGEMENTS

We thank dr. Jovan Tanevski for his initial work on the project, as well as his ideas and help in further work.

REFERENCES

- [1] Jure Brence, Jovan Tanevski, Jennifer Adams, Edward Malina, and Sašo Džeroski. 2020. Learning surrogates of a radiative transfer model for the sentinel 5p satellite. In *Proceedings of International Conference on Discovery Science (Lecture Notes in Computer Science)*. Volume 12323.
- [2] A Butz, André Galli, O Hasekamp, J Landgraf, P Tol, and I Aben. 2012. Tropomi aboard sentinel-5 precursor: prospective performance of ch4 retrievals for aerosol and cirrus loaded atmospheres. *Remote Sensing of Environment*, 120, 267–276.
- [3] David Charte, Francisco Charte, Salvador García, María J del Jesus, and Francisco Herrera. 2018. A practical tutorial on autoencoders for nonlinear feature fusion: taxonomy, models, software and guidelines. *Information Fusion*, 44, 78–96.
- [4] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. 2001. *The elements of statistical learning*. Number 10. Volume 1. Springer series in statistics New York.
- [5] Pierre Geurts, Damien Ernst, and Louis Wehenkel. 2006. Extremely randomized trees. *Machine learning*, 63, 1, 3–42.
- [6] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. <http://www.deeplearningbook.org>. MIT Press.
- [7] Otto P Hasekamp and J Landgraf. 2002. A linearized vector radiative transfer model for atmospheric trace gas retrieval. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 75, 2, 221–238. ISSN: 00224073. DOI: 10.1016/S0022-4073(01)00247-3.
- [8] Haili Hu, Otto Hasekamp, André Butz, André Galli, Jochen Landgraf, Joost Aan de Brugh, Tobias Borsdorff, Remco Scheepmaker, and Ilse Aben. 2016. The operational methane retrieval algorithm for tropomi. *Atmospheric Measurement Techniques (AMT)*, 9, 11, 5423–5440.
- [9] IPCC. 2014. Fifth Assessment Report - Impacts, Adaptation and Vulnerability. (2014). Retrieved 06/12/2017 from <http://www.ipcc.ch/report/ar5/wg2/>.
- [10] Andy Liaw, Matthew Wiener, et al. 2002. Classification and regression by randomforest. *R news*, 2, 3, 18–22.
- [11] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9, Nov, 2579–2605.
- [12] Leland McInnes, John Healy, and James Melville. 2018. Umap: uniform manifold approximation and projection for dimension reduction, (December 2018). <https://arxiv.org/abs/1802.03426>.
- [13] J Tanevski, S Džeroski, and T Todorovski. 2019. Meta-model framework for surrogate-based parameter estimation in dynamical systems. *IEEE Access*, 99.

Deep Multi-label Classification of Chest X-ray Images

Dejan Štepec
dejan.stepec@xlab.si

University of Ljubljana, Faculty of Computer and Information Science
XLAB d.o.o.
Ljubljana, Slovenia

ABSTRACT

In this paper we address the problem of Chest X-ray (CXR) classification in a multi-label classification (MLC) setting, in which each sample can be associated with one or several labels. The availability of large-scale CXR datasets has provided the ability to develop highly accurate deep-learning based supervised models, that closely resembles the performance of human radiologists. We compare an end-to-end deep-learning based approach with different ensembles of predictive clustering trees (PCTs) and show that similar predictive performance can be achieved, when using the features extracted from the pre-trained deep-learning model.

KEYWORDS

Chest X-ray, deep-learning, predictive clustering trees, random forest, extra tree

1 INTRODUCTION

Chest X-ray (CHR) is one of the most common medical imaging modalities, with millions of scans performed globally every year [6]. A computer-aided diagnosis (CAD) system can significantly reduce the burden of radiologists and thus reduce prevalence and early detection of many deadly diseases. There has been a lot of effort recently, to harness the power of machine learning based methods, especially deep-learning, for disease classification and localization from CXR images [17]. Interpreting CXR images is very difficult even for the trained pathologists, with different visual ambiguities representing a significant challenge to distinguish between different diseases, resulting in misdiagnoses [5].

Recently, deep-learning based approaches have been presented, that together with the availability of large-scale datasets significantly improve the performance of CAD methods and in some cases reach the radiologist-level performance [8]. In comparison with other approaches and datasets [9, 13, 1], newly presented datasets [8, 10] enable the development of CAD methods for detection of presence of multiple diseases present in CXR images at the same time.

We evaluate an end-to-end deep-learning based approach for multi-label classification (MLC) of CXR images, based on DenseNet architecture [7] and compare it with the traditional approach based on predictive clustering trees (PCT) [2], in an ensemble setting, using the features extracted from the pre-trained deep-learning network. We demonstrate a similar predictive performance on a large-scale CheXpert dataset [8], thus opening the potential to use PCTs also in a hierarchical setting [20], which

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

Information society '20, October 5–9, 2020, Ljubljana, Slovenia

© 2020 Copyright held by the owner/author(s).

taking into account underlying dependency structure and powerful deep features, could advance current state-of-the-art of the supervised MLC deep-learning based approaches.

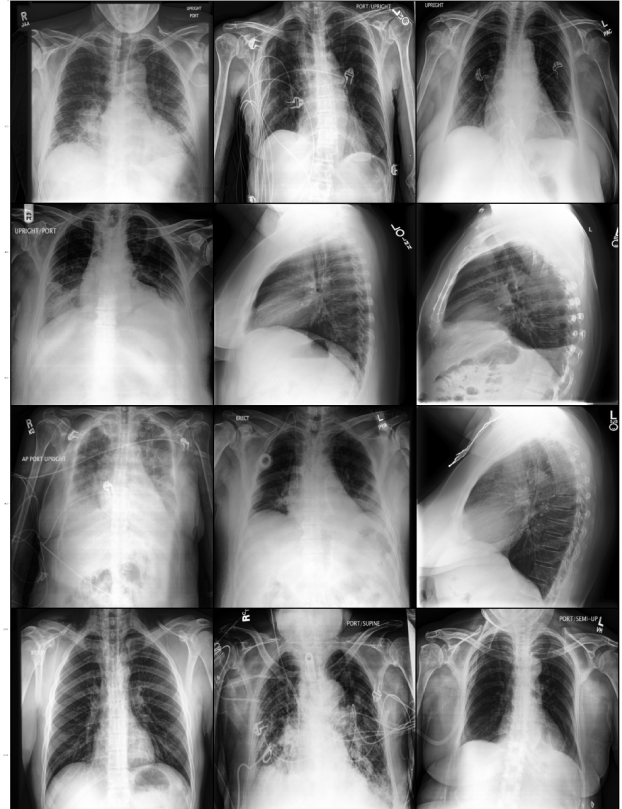


Figure 1: Few examples of Chest X-ray images from the CheXpert dataset [8].

2 RELATED WORK

Recent prevalence of deep-learning methods and increased availability of large-scale datasets with labeled data has provided medical community with significant advances, in comparison with the methods that require sub-optimal manual feature engineering [14]. State-of-the-art CNN models are becoming a de-facto standard for a wide range of application in medical imaging, such as detection, classification and segmentation. Similar advances in terms of the methods and available data have been observed in the domain of Chest X-ray (CXR) images.

Multi-label classification (MLC) setting is a very common setting in interpreting CXR images, due to presence of multiple diseases in one particular CXR sample. Deep-learning architecture CheXNet [19] was proposed, based on DenseNet-121 [7], trained on ChestX-ray14 dataset [21], which achieved state-of-the-art results over 14 labeled pathologies and even exceeded

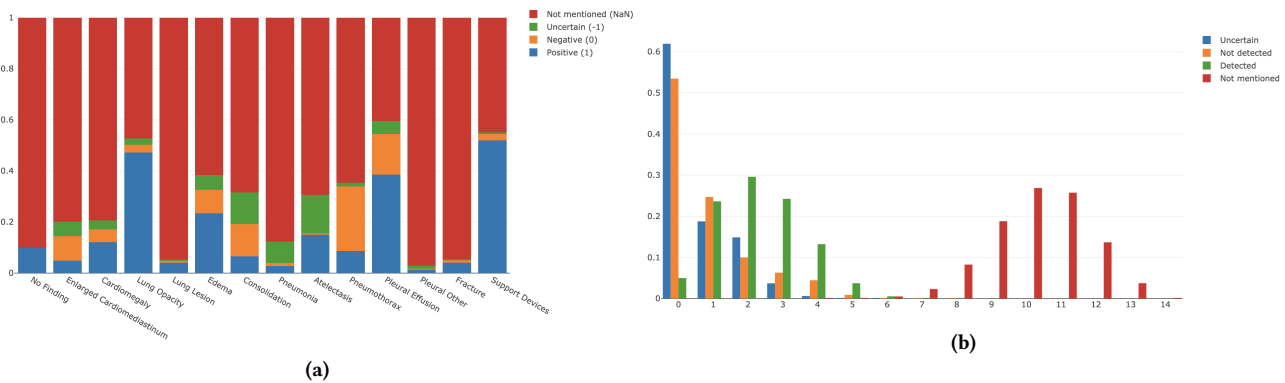


Figure 2: (a) Label uncertainty distribution over 14 pathologies in the CheXpert dataset [8] over all the samples in the training data and (b) distribution and probability of occurrence of multiple pathologies in a particular sample (multi-label classification).

radiologist performance on pneumonia. Recently, very large-scale CXR datasets were presented, such as CheXpert [8] and MIMIC-CXR [10], which enabled the development of much more robust supervised models. Additionally, the new datasets also capture the notion of uncertainty through labels and different approaches have been proposed for handling such labels. A similar architecture to CheXNet was proposed and performance surpassed 3 certified radiologists in 3 different pathologies [8].

The above MLC approaches do not take into the account the dependencies between disease labels, which, when exploited, significantly improves the performance of the predictive models [16]. We evaluate an end-to-end deep-learning based approach for MLC of CXR images, based on DenseNet architecture [7] and compare it with the traditional approach based on predictive clustering trees (PCTs) [2], in an ensemble setting, using the features extracted from the pre-trained deep-learning network. We demonstrate a similar predictive performance on a large-scale CheXpert dataset [8], thus opening the potential to use PCTs also in a hierarchical setting [20], which taking into account underlying dependency structure and powerful deep features, could advance current state-of-the-art of the supervised MLC deep-learning based approaches and also compete against hierarchical deep-learning based approaches [4, 16], which take the hierarchy into account implicitly, using the conditional probability.

3 CHEXPART: A LARGE CHEST RADIOGRAPH DATASET

CheXpert [8] is a large publicly available dataset for chest radiograph interpretation, consisting of 224,316 CXR images of 65,240 patients, where the presence of 14 different observations is labeled as positive, negative, uncertain or not mentioned. CXR images are collected retrospectively from Stanford Hospital, together with associated radiology reports. Labels (and their uncertainty) were automatically extracted from the section of the radiology report, which summarizes the key findings. A large list of phrases was manually curated by multiple board-certified radiologists to match various ways of observations, mentioned differently in the reports. Extracted phrases are then classified into positive, negative, uncertain or not-mentioned classes and aggregated into a final set of predefined observations (i.e. pathologies) with prevailed occurrence. The publicly available test data consists of 234 samples from 234 patients, where ground truth is set by a consensus of 3 radiologists, who annotated the set

using radiographs, thus labels only represent positive or negative class, with no uncertainties. Evaluation is performed only on 5 observations, selected based on their clinical significance and prevalence in the dataset (i.e. Atelectasis, Cardiomegaly, Consolidation, Edema and Pleural Effusion).

The distribution of all the observed pathologies in the training data and their uncertainty is presented in Figure 2a and the distribution of observations over a single example in Figure 2b, which shows that there is around 30% chance of having at least 2 pathologies present at the same time, labeled as definite positive. In CheXpert [8], different strategies of using uncertainty labels were evaluated. The two most simple approaches are to ignore uncertain samples during the training or to map them to either negative or positive class. They also evaluate a semi-supervised approach, where the ignore approach is used to label uncertain examples, in order to re-label them. 3-class classification approach is also evaluated where uncertain label is used as a separate class during the training and during testing, only the probabilities for positive and negative class are reported. In our work, we use the simple mapping approach, by mapping uncertain labels to a positive class and not-mentioned samples to a negative class.

3.1 Methods

We evaluate an end-to-end deep-learning based approach for multi-label classification (MLC) of CXR images, based on DenseNet-121 architecture [7] and compare it with the traditional approach based on predictive clustering trees (PCT) [2], in an ensemble setting, using the features extracted from the pre-trained deep-learning network.

3.2 End-To-End Deep Learning

Several convolutional neural networks (CNNs) were evaluated in CheXpert [8] and DenseNet-121 [7] architecture produced the best results. Because of that, we used DenseNet-121 for all of our experiments. Original DenseNet is designed for multi-class classification, where the neural network has the same number of output nodes as the number of classes. Each output node belongs to some class and outputs a score for that class. In a multi-class setting, the scores are passed through softmax layer, which converts scores into probabilities (class probabilities sums to 1) and the input sample is classified into a corresponding class, that has the highest probability.

In a multi-label classification (MLC) setting, the difference is, that an input sample can belong to multiple classes at the same time, thus the final score needs to be independent for each of the classes, because of that, sigmoid function is used instead of softmax. Additionally, categorical cross-entropy loss function needs to be replaced with binary cross-entropy. We implemented modified DenseNet-121 in PyTorch¹ using Adam optimizer with the same learning rates and parameters as used in CheXpert [8]. The images were resized to 320 x 320, same as in [8] and we trained the network for 10 epochs using a fixed batch size of 32 images and evaluated the performance on a left-out validation set of 500 images using the receiver operating characteristic curve (ROC) and its area under the curve (AUC), averaged across all observations. The best performing model in terms of global AUC score was selected for evaluation on a test set, presented in Section 4.

3.3 Predictive Clustering Trees

Predictive clustering trees (PCTs) [2] are decision trees viewed as a hierarchy of clusters, where the top node corresponds to one cluster containing all the data, which is recursively partitioned into smaller clusters while moving down the tree. PCTs are constructed with a standard "top-down induction of decision trees" (TDIDT) algorithm, the major difference in comparison with CART [3] or C4.5 [18] induction is that the PCTs treat variance and prototype functions as parameters, selected based on the learning task at hand. To construct a regression tree, for example, the variance function returns the variance of the given instances' target values, and the prototype is their average value. For the task of predicting tuples of discrete variables, used in the multi-label classification (MLC) [15], the variance functions is computed as the sum of the Gini indices[28] of the variables from the target tuple and the prototype function returns a vector of probabilities, that an example belongs to a particular class in the target tuple.

In our work we utilized PCTs in an ensemble setting, where a set of predictive models (i.e. PCTs) predictions are combined to obtain a final prediction, this is especially useful for unstable base predictors (e.g. trees), where small changes in the dataset, yield substantially different models and usually achieves a much better predictive performance [12]. In our work we consider a Random forest of PCTs (RF-PCT) [12] and ensembles of extremely randomized PCTs (EXTRA-PCT) [11] for MLC. In RF-PCT, several bootstrap replicates are first constructed and a randomized PCT is then applied, by selecting a subset of attributes in each node, on which all possible tests are considered and the best one is selected. The number of attributes selected is a given parameter, typically a function of the total number of attributes (e.g. $\log(N)$ - where N represents the number of attributes). In EXTRA-PCT, no bootstrap replicates are constructed and in each internal node, for the each attribute, a test is selected randomly.

We used CLUS² framework for the PCT construction. We used 50 baseline PCTs for RF-PCT, as well as EXTRA-PCT. The input presented the 1024D features extracted from the pre-trained DenseNet-121 network, extracted before the last fully-connected classification layer in DensetNet-121. Similarly to the DenseNet-121 end-to-end approach, the RF-PCT and EXTRA-PCT were evaluated on a test set in terms of AUC score.

¹https://pytorch.org/hub/pytorch_vision_densenet

²<http://clus.sourceforge.net/>

4 RESULTS

We evaluated different approaches on the publicly available test data, consisting out of 234 samples from 234 patients, where ground truth is set by a consensus of 3 radiologists. We report the results in terms of the Receiver Operating Characteristic Curves (ROC) in Figure 3 and its area under the curve (AUC) in Table 1. In terms of the approaches presented in our work (i.e. DenseNet-121, RF-PCT and EXTRA-PCT), DenseNet-121 performs the best, with EXTRA-PCT approach following it closely. The biggest differences are observed on the Cardiomegaly class, which coincides with the results reported in CheXpert [8], as most of the uncertain cases are borderline, which reduces the performance of the simple mapping to positive or negative label.

Table 1 also compares presented approaches against the DenseNet-121 baseline presented in CheXpert [8], where 10 checkpoints per run were chosen and each model was run three times, thus generating an ensemble of 30 models, which improved the results by a small margin over our baseline DenseNet-121 approach. Nevertheless, we achieved or surpassed CheXpert results on Cardiomegaly and Pleural Effusion classes and also achieved similar performance on other classes.

5 CONCLUSION

In this paper we addressed the problem of Chest X-ray (CXR) classification in a multi-label classification (MLC) setting and compared an end-to-end deep-learning based approach with different ensembles of predictive clustering trees (PCTs) and showed that similar predictive performance can be achieved, when using the features extracted from the pre-trained deep-learning model. This results show the potential to use PCTs also in a hierarchical setting, which taking into account underlying dependency structure and powerful deep features, could advance current state-of-the-art.

ACKNOWLEDGMENTS

This work has been supported by the H2020 iPC project (826121)

REFERENCES

- [1] Worawate Ausawalaithong, Arjaree Thirach, Sanparith Marukatat, and Theerawit Wilaiprasitporn. 2018. Automatic lung cancer prediction from chest x-ray images using the deep learning approach. In *2018 11th Biomedical Engineering International Conference (BMEiCON)*. IEEE, 1–5.
- [2] Hendrik Blockeel, Luc De Raedt, and Jan Ramon. 1998. Top-down induction of clustering trees. In *Proceedings of the Fifteenth International Conference on Machine Learning (ICML '98)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 55–63.
- [3] Leo Breiman, JH Friedman, RA Olshen, and CJ Stone. 1984. Classification and regression trees. statistics/probability series. (1984).
- [4] Haomin Chen, Shun Miao, Daguang Xu, Gregory D Hager, and Adam P Harrison. 2019. Deep hierarchical multi-label classification of chest x-ray images. In *International Conference on Medical Imaging with Deep Learning*, 109–120.
- [5] Louke Delrue, Robert Gosselin, Bart Ilsen, An Van Landeghem, Johan de Mey, and Philippe Duyck. 2011. Difficulties in the interpretation of chest radiography. In *Comparative interpretation of CT and standard radiography of the chest*. Springer, 27–49.

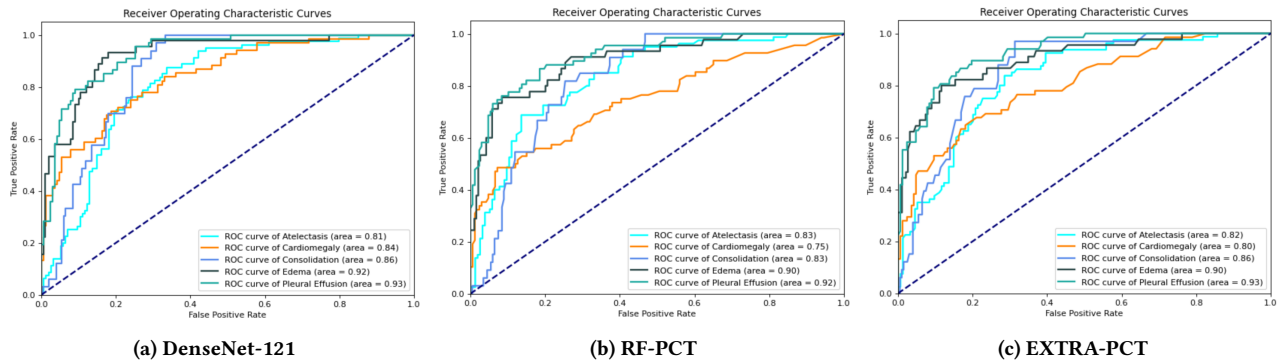


Figure 3: Receiver Operating Characteristic Curves (ROC) for an end-to-end deep-learning approach based on DenseNet-121 (a) and ensemble of Predictive Clustering Trees (PCTs) based on random forest (b) and extremely randomized trees (c).

Table 1: Comparison of different methods against the baseline CheXpert results [8] in terms of AUC scores.

Method	Atelectasis	Cardiomegaly	Consolidation	Edema	Pleural Effusion
CheXpert (U-Ones) [8]	0.86	0.83	0.90	0.94	0.93
DenseNet-121	0.81	0.84	0.86	0.92	0.93
RF-PCT	0.83	0.75	0.83	0.90	0.92
EXTRA-PCT	0.82	0.80	0.86	0.90	0.93

- [6] [n. d.] Diagnostic Imaging Dataset 2019-20 Data, NHS England. (Accessed 4 August 2020).
- [7] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. 2017. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4700–4708.
- [8] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silvana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. 2019. Chexpert: a large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI Conference on Artificial Intelligence*. Volume 33, 590–597.
- [9] Amit Kumar Jaiswal, Prayag Tiwari, Sachin Kumar, Deepak Gupta, Ashish Khanna, and Joel JPC Rodrigues. 2019. Identifying pneumonia in chest x-rays: a deep learning approach. *Measurement*, 145, 511–518.
- [10] Alistair EW Johnson, Tom J Pollard, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Yifan Peng, Zhiyong Lu, Roger G Mark, Seth J Berkowitz, and Steven Horng. 2019. MIMIC-CXR-JPG, a large publicly available database of labeled chest radiographs. *arXiv preprint arXiv:1901.07042*.
- [11] Dragi Koccev and Michelangelo Ceci. 2015. Ensembles of extremely randomized trees for multi-target regression. In *International Conference on Discovery Science*. Springer, 86–100.
- [12] Dragi Koccev, Celine Vens, Jan Struyf, and Sašo Džeroski. 2013. Tree ensembles for predicting structured outputs. *Pattern Recognition*, 46, 3, 817–833.
- [13] Paras Lakhani and Baskaran Sundaram. 2017. Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks. *Radiology*, 284, 2, 574–582.
- [14] June-Goo Lee, Sanghoon Jun, Young-Won Cho, Hyunna Lee, Guk Bae Kim, Joon Beom Seo, and Namkug Kim. 2017. Deep learning in medical imaging: general overview. *Korean journal of radiology*, 18, 4, 570–584.
- [15] Gjorgji Madjarov, Dragi Koccev, Dejan Gjorgjevič, and Sašo Džeroski. 2012. An extensive experimental comparison of methods for multi-label learning. *Pattern recognition*, 45, 9, 3084–3104.
- [16] Hieu H Pham, Tung T Le, Dat Q Tran, Dat T Ngo, and Ha Q Nguyen. 2019. Interpreting chest x-rays via cnns that exploit disease dependencies and uncertainty labels. *arXiv preprint arXiv:1911.06475*.
- [17] Chunli Qin, Demin Yao, Yonghong Shi, and Zhijian Song. 2018. Computer-aided detection in chest radiography based on artificial intelligence: a survey. *Biomedical engineering online*, 17, 1, 113.
- [18] J Ross Quinlan. 2014. *C4.5: programs for machine learning*. Elsevier.
- [19] Pranav Rajpurkar, Jeremy Irvin, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Ding, Aarti Bagul, Curtis Langlotz, Katie Shpanskaya, et al. 2017. Chexnet: radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv preprint arXiv:1711.05225*.
- [20] Celine Vens, Jan Struyf, Leander Schietgat, Sašo Džeroski, and Hendrik Blockeel. 2008. Decision trees for hierarchical multi-label classification. *Machine learning*, 73, 2, 185.
- [21] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. 2017. Chestx-ray8: hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2097–2106.

Smart Issue Retrieval Application

Jernej Zupančič
jernej.zupancic@ijs.si
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia

Borut Budna
borut.budna@ijs.si
Faculty of Computer and
Information Science
Ljubljana, Slovenia

Miha Mlakar
Maj Smerkol
miha.mlakar@ijs.si
maj.smerkol@ijs.si
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia

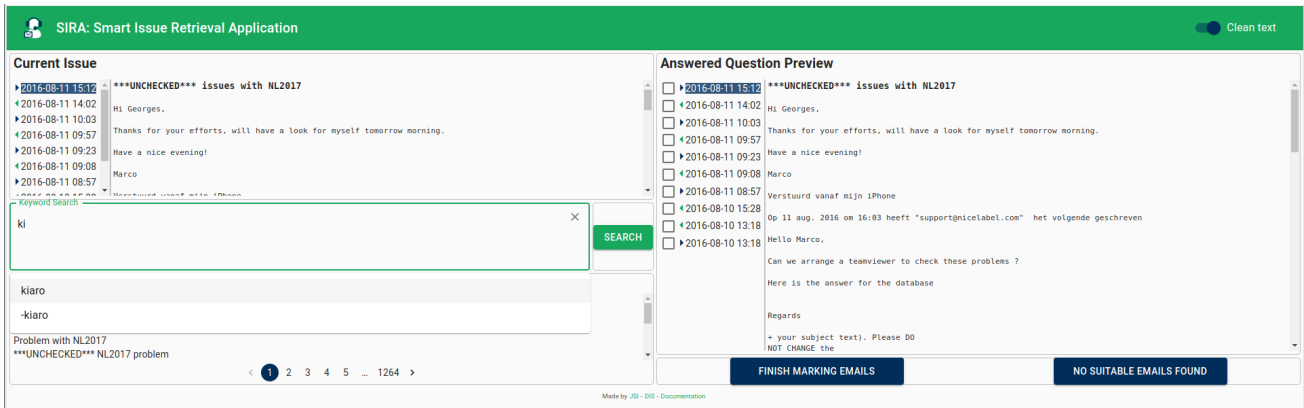


Figure 1: SIRA screenshot

ABSTRACT

We present Smart Issue Retrieval Application (SIRA), a customer support tool for searching of relevant email threads or issues when an email thread and keywords are given. Presented are the overall application architecture, the processing pipeline, which transforms the data into a search friendly form, and the search algorithm itself.

KEYWORDS

customer support, language models, information retrieval

1 INTRODUCTION

Customer support is an important part of many large businesses and high quality customer support can improve the user experience and help businesses retain their customer for longer periods. For larger companies, it can also be a strain on their human resources as many customer support issues need to be resolved in short time. While the customer support team may resolve most issues on their own sometimes they need the help of the development department. Often similar issues are presented to the developers multiple times.

In order to minimize the number of issues that need attention from other departments, we have developed an application to help the customer support technicians resolve issues without help from developers. While some issues will still need the attention of

developers, SIRA can help find existing answers to questions that have already been resolved by developers and therefore reduce the amount of distractions for the development team.

We use language models in order to retrieve information about the question from the issue at hand. Using multiple different approaches, application searches the database of resolved issues in order to find a developers' answers to same or similar questions.

2 SIRA ARCHITECTURE

SIRA comprises five main application components (Fig. 2):

- (1) *Database*. PostgreSQL [6] is used as the application database, since it includes decent built-in text search capabilities and change data capture options.
- (2) *Processing daemon*. Python [7] process responsible for data processing for search in the event of change data capture.
- (3) *Back-end application*. Python Flask-based back-end application exposing the application programming interface for SIRA.
- (4) *Front-end application*. React-based [8] single-page application for interacting with SIRA.
- (5) *Documentation*. MKdocs-based user documentation for final users, admins, and developers.

Each SIRA component is packaged within a Docker [4] image and can be managed using “docker-compose” [2] tool. This enables deterministic packaging of application code for development, testing and production.

3 SIRA FUNCTIONALITY

The main goal of SIRA is to enable customer support staff to quickly find answers to similar questions that have already been resolved in the past. Search is therefore the primary functionality of the application and can be split into three parts:

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

Information society '20, October 5–9, 2020, Ljubljana, Slovenia

© 2020 Copyright held by the owner/author(s).

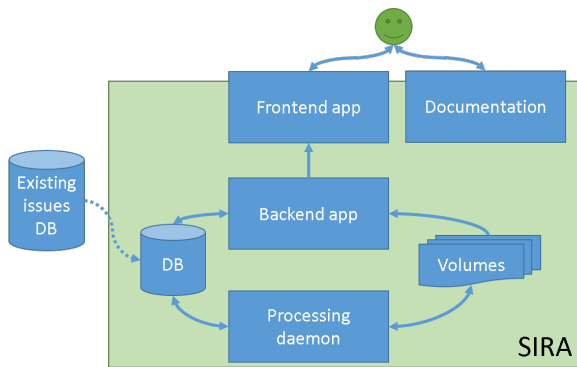


Figure 2: SIRA architecture overview

- (1) *Processing*. Upon new data arrival, pre-processes the text to obtain representation suitable for search.
- (2) *Search*. Computing relevancy scores upon search request by taking into account as much information about issue or email thread as possible.
- (3) *Logging*. To improve the search in the future the search results and structured user feedback is gathered and stored.

In the rest of this section we will describe each part in more details.

3.1 Processing

For the search to be efficient it is beneficial to pre-process the raw emails. The processing daemon runs as a separate python process and utilizes PostgreSQL’s logical replication functionality in order to transform new content as soon as it is written to the database. The following steps are executed when processing the issues:

- (1) *HTML clean*. BeautifulSoup [1] library is used to extract only relevant text from email XML markup.
- (2) *Empty line removal*. Python script is used to detect and remove empty lines.
- (3) *Repeated emails removal*. Parts of emails are deleted if they already appear within some previous mails of the same issue.
- (4) *Semi-structured emails handling*. Some emails are actually a filled out form in an email format. A python script is used to extract only the relevant information.
- (5) *Non-author lines removal*. A machine learning model was developed and is deployed for tackling this task.
- (6) *Non-alphanumeric-only characters lines removal*. Python script is used to detect and remove those lines.
- (7) *Word vector representation computation and update*. FastText [3] word vectors are used to compute word vector representation of text.
- (8) *Storing of processed text*. The processed text is stored into database, where built-in database indexing is utilized to further prepare the text for efficient text searching.

In the rest of this section we focus on the non-trivial processing steps.

3.1.1 Repeated emails removal. There were two reasons for removing repeated emails from an email thread. First, when displaying an email, usually also all the previous emails are included, which results in poor readability. Second, some methods for comparing the text take into account the number of occurrences of a

particular word. This is sensible for cases when the word actually repeats in the content. However, if it repeats due to the text duplication it could negatively impact the search results.

We define a repeated email as an email body that appears within another email body. This is usually a result of using a “Reply” functionality when responding to an email within an email client.

To delete email A from email B , the following method is used:

- (1) Extract only alphanumeric characters from the two email bodies A and B to get $\text{alphanumeric}(A)$ and $\text{alphanumeric}(B)$.
- (2) If $\text{alphanumeric}(A)$ appears within $\text{alphanumeric}(B)$, mark it for removal from $\text{alphanumeric}(B)$.
- (3) If $\text{alphanumeric}(A)$ does not appear within $\text{alphanumeric}(B)$, iterate over substrings of $\text{alphanumeric}(B)$ and compute the matching percentage of consecutive alphanumeric blocks from $\text{alphanumeric}(A)$. The substring with the maximum match is a candidate for removal. If it exceeds a predefined threshold it is indeed marked for removal from $\text{alphanumeric}(B)$.
- (4) Reconstruct B by dropping the substring marked for removal and all non-alphanumeric characters positioned within the marked substring when expanded with all the characters.

3.1.2 Non-author lines removal. An email body usually comprises:

- (1) Relevant content
- (2) Signature
- (3) Confidentiality notice
- (4) Previous email headers
- (5) Previous email content

The only text that should be used for text comparison is the relevant content part. While previous email content was mostly removed in the repeated emails removal step (3.1.1), other email body parts can still impact text comparison results. Machine learning was utilized to develop a model for determining whether a particular line in the email body belongs to the relevant content part of an email or not.

Dataset preparation. First, we implemented an application with a basic graphical user interface that enabled us to label each line with one of the following categories:

- (1) AUTHOR. The relevant content falls into this category.
- (2) QUOTED. This is the previous email content.
- (3) AUTO-PERSONALIZED. This is the text, that was set by a user in the email client, which is automatically inserted by the email client. Signature is an example of this.
- (4) AUTO-NON-PERSONALIZED. This is the text inserted by the email client automatically. An example of this is previous email headers.
- (5) NEEDS-PRETTIFY. Sometimes the whole email body is present in one line only. To properly label the body it should be further split into multiple lines.
- (6) OTHER. Everything else.

Second, we labeled each line belonging to 100 random issues. This way we generated a dataset of 37,421 labeled lines in 586 emails. Since the assumption was that the “QUOTED” lines are already filtered out using remove repeated emails method, we omit those lines from the dataset. This left us with 9,848 labeled lines.

Features. The computed features were of two types: local features that took into account just the current line, and global

features that took into account the relative position and content of a line within the whole email.

Local features:

- (1) Number and proportion of capitalized words
- (2) Number and proportion of non-alphanumeric characters
- (3) Number and proportion of numeric characters
- (4) “CountVectorizer” from the scikit-learn ([5]) package
- (5) “TfidfVectorizer” from the scikit-learn package
- (6) Word vector line representation

Global features:

- (1) Line position from the start
- (2) Line position until the end
- (3) Does “regard” appear before this line, within this line, after this line
- (4) Do four or more consecutive non-alphanumeric characters appear before this line, within this line, after line
- (5) Does a date-like string appear before this line, within this line, after this line
- (6) Does a time-like string appear before this line, within this line, after this line

In order to smooth the predictions we also tested hierarchical modeling by first building a model for “AUTHOR” detection and then using the predictions on the lower level as additional features for the higher level. One approach for using the predictions from the lower level was to just use the “AUTHOR” predictions of lines just before and just after the current line. The predictions were padded with 1 at the beginning of an email and with 0 at the end. The second approach was based on the sum of three consecutive “AUTHOR” class probabilities for: lines, just before the current line, lines where the current line is in the middle, and lines just after the current line. We padded the predictions with 1s at the beginning of an email and with 0s at the end.

Further, the features were scaled using the StandardScaler and the feature space dimensionality was reduced using the principal component analysis - PCA, both from the scikit-learn package.

Models. For modeling we utilized scikit-learn package and tested the following algorithms: (1) Logistic regression, (2) Multinomial Naive Bayes, (3) Support vector machine, (4) Random forest classifier.

Rudimentary hyper-parameter tuning was done to pick the best ones.

Evaluation. Each pipeline was evaluated using 10-fold cross validation with the splits over issues. This means that all the lines belonging to one issue were either in the training or the testing set to prevent data leaking.

Model selection. The performance of all models was tracked through various metrics:

- (1) Confusion matrix
- (2) Precision and recall at different minimum recall thresholds
- (3) Precision-recall curve
- (4) “AUTHOR” probabilities for each line in the test set

The main concern regarding the model performance was that it should prioritize keeping the “AUTHOR” lines (“AUTHOR” recall) over average model accuracy. This is a direct result of the application architecture – if the line would be removed by the chosen model, it wouldn’t be possible to search over it. This would directly impact the performance in the real-world. Additionally, few additional lines shouldn’t hinder the readability too much.

The gathered metrics enabled us to closely inspect each model and overview the performance regarding real-world application.

A basic GUI was built to inspect the models and overview the miss-classified examples. In the end, the hierarchical model was chosen with most of the presented features, with the exception of “CountVectorizer” and “TfidfVectorizer” features. The additional chosen higher-level feature was the sum of three consecutive “AUTHOR” probabilities. Random forest was chosen as the classification algorithm, without feature standardization or dimensionality reduction step. The threshold probability was lowered to 0.12 so recall could be kept high.

The final model miss-classified 59 out of 2,394 rows marked as “AUTHOR” (recall = 0.975) and 629 out of 7,454 rows marked as “OTHER” (recall = 0.806).

3.1.3 Word vector representation computation and update. Word vector representation of content is used to compare email bodies and email subjects between different issues.

To compute the word vector representation of text, either issue body or issue subject, the following steps are executed: (1) Tokenize text, (2) Remove stop-words, (3) Query word vector representation for each word using fastText common crawl word vectors with dimension 300, (4) Compute mean of all word vectors belonging to the words in the text, (5) Normalize the mean vector by dividing the mean vector by the mean vector length.

Instead of generating the representation vectors on-the-fly, they are pre-computed and only read when needed, which greatly reduces the inference time. To update word vector representation of a particular text, the corresponding row in the word vector matrix is updated with the new values and stored on disk as a Numpy array.

3.2 Search

Each issue consists of: subject, document (the email body of text), and keywords the user marked the issues with. The keywords can be positive, meaning that a keyword is related with the contents of the issue, or negative when keyword is *not* related with the contents of the particular issue. Additionally, a keyword can be explicit, where a user uses the keyword for searching when considering a particular issue. On the other hand, a keyword can be implicit – soft keywords, where the user searched for relevant issues using a keyword, but the search results were not marked as relevant.

When computing the relevancy of issues, given a starting issue and some keywords, several relevancy sub-scores are first computed and then aggregated to form a single relevancy score. In Table 1 all combinations for relevance sub-scores are listed.

The final score is computed as a weighted average, as in equation 1. The weights w_i were determined based on the final user feedback.

$$\begin{aligned}
 \text{finalScore} = & w_1 \cdot \text{KeywordToKeywordScore} \\
 & + w_2 \cdot \text{KeywordToSoftKeywordScore} \\
 & + w_3 \cdot \text{KeywordToDocumentScore} \\
 & + w_4 \cdot \text{KeywordToSubjectdScore} \\
 & + w_5 \cdot \text{DocumentToKeywordScore} \quad (1) \\
 & + w_6 \cdot \text{DocumentToSoftKeywordScore} \\
 & + w_7 \cdot \text{DocumentToDocumentScore} \\
 & + w_8 \cdot \text{SubjectToKeywordScore} \\
 & + w_9 \cdot \text{SubjectToSoftKeywordScore} \\
 & + w_{10} \cdot \text{SubjectToSubjectScore}
 \end{aligned}$$

Table 1: Relevance sub-scores matrix

		Other issues			
		(Not) Keyword	Soft (Not-) keyword	Document	Subject
Current issue	(Not-) Keyword	Exact match	Exact match	Full-text search	Full-text search
	Soft (Not) Keyword	/	/	/	/
	Document	Reverse full-text search	Reverse full-text search	Word vector cosine similarity	/
	Subject	Reverse full-text search	Reverse full-text search	/	Word vector cosine similarity

3.2.1 Exact match. This relevance score compares (soft) keywords related to issues and those inserted in the keyword input box. Given a (soft) keyword, search for all the documents that are in relation to this exact (soft) keyword. Each relation can either be positive or negative. Therefore, the returned score is positive in case of positive relation and negative otherwise.

3.2.2 Full-text search. This relevance score compares keywords entered in the keyword input box and issue documents or issue subjects. Full-text search capability of PostgreSQL is leveraged for this score. However, the results are modified to return negative scores in case of not-keyword match.

3.2.3 Reverse full-text search. This relevance score compares the selected issue document or subject and all existing (soft) keywords. First, for each keyword a full-text search relevance score is computed. Second, for each issue in the database do a sum of its related keyword relevance scores.

3.2.4 Word vector cosine similarity. This relevance score compares the selected issue document and subject to all existing issue documents and subjects, respectively. Pre-computed word vectors as described in Section 3.1.3 are used. The relevance score is computed as:

$$\text{wordVectorSimilarity}(T_1, T_2) = 1 - T_1 \cdot T_2. \quad (2)$$

Since the word vectors used are normalized, this is actually $1 - \text{cosine distance between } T_1 \text{ and } T_2$.

Two other methods for comparing the text were also tested: PostgreSQL built-in trigram text similarity, which was too slow for production use, and tf-idf representation of text and cosine distance-based relevance score, which did not perform as well as the word vectors method.

3.3 Logging

To improve the search performance in the future, several interactions with the application are logged:

- (1) Search results with relevance scores
- (2) Viewed search results
- (3) Relevant issue/belonging email found
- (4) No relevant issue/belonging email found

Only after sufficient real-world usage of the application we can quantitatively evaluate the performance of the whole search pipeline and act upon the results.

4 DISCUSSION AND CONCLUSION

The SIRA system was developed and deployed, including five docker-image packaged modules. The main functionalities of the first major release include preprocessing of the text of the issue, search integrating four different search algorithms and a logging system that stores interactions with the system into

the database, including user defined keywords and appropriate results marking.

Preprocessing is done without any user interaction and involves multiple algorithms and AI methods to extract the text of the issue from original encoded emails. Testing of the algorithms shows good results both in terms of precision and recall. Word vector representations are pre-computed in order to improve performance of search algorithms.

Based on the extracted plain text of the issue the application searches for similar issues that have already been resolved. The users can therefore quickly find the information related to the issue. The system is currently in use and only after some time of real-world usage we will be able to evaluate the whole system.

Due to logging the interactions in the database we expect to be able to analyze the usage and quality of the results. This will allow us to improve the system and add other functionality that will improve user experience and further improve the customer support technicians' workflow.

ACKNOWLEDGMENTS

Nicelabel d.o.o. funded the research presented in this paper. We thank Gregor Grasselli, Zdenko Vuk and Miha Štravs for help in application development.

REFERENCES

- [1] Beautiful Soup Developers. 2019. Beautiful soup. <https://www.crummy.com/software/BeautifulSoup/>. (2019).
- [2] Docker Inc. 2019. Docker-compose. <https://docs.docker.com/compose/>. (2019).
- [3] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.
- [4] Dirk Merkel. 2014. Docker: lightweight linux containers for consistent development and deployment. *Linux journal*, 2014, 239, 2.
- [5] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
- [6] PostgreSQL Global Development Group. 2019. PostgreSQL, version 12. <http://www.postgresql.org>. (2019).
- [7] Python Software Foundation. 2018. Python language reference, version 3.7. <http://www.python.org>. (2018).
- [8] React Developers. 2019. React. <https://reactjs.org/>. (2019).

Adaptation of Text to Publication Type

Luka Žontar

University of Ljubljana,
Faculty of Computer and Information Science
Ljubljana, Slovenia
zontarluka98@gmail.com

Zoran Bosnić

University of Ljubljana,
Faculty of Computer and Information Science
Ljubljana, Slovenia
zoran.bosnic@fri.uni-lj.si

ABSTRACT

In this paper, we propose a methodology that can adapt texts to target publication types using summarization, natural language generation and paraphrasing. The solution is based on key text evaluation characteristics that describe different publication types. To examine types, such as social media posts, newspaper articles, research articles and official statements, we use three distinct text evaluation metrics: length, text polarity and readability. Our methodology iteratively adapts each of the text evaluation metrics. To alter length, we focus on abstractive summarization using text-to-text transformers and distinct natural language generation models that are fine-tuned for each target publication type. Next, we adapt polarity and readability using synonym replacement and additionally, manipulate the latter by replacing sentences with paraphrases, which are automatically generated using a fine-tuned text-to-text transformer. The results show that the proposed methodology successfully adapts text evaluation metrics to target publication types. We find that in some cases adapting the chosen text evaluation metrics is not enough and we can corrupt the content using our methodology. However, generally, our methodology generates suitable texts that we could present to a target audience.

KEYWORDS

text adaptation, context-aware, artificial intelligence, text summarization, natural language processing

1 INTRODUCTION

With more and more internet usage, the textual data on the internet is highly increasing. However, different media target different audiences and thus an arbitrary article may not be appropriate for everyone. Consequently, already published content is being rewritten and adapted for other target audiences.

Why is targeting audiences so important? When speaking with someone in person, we adjust body language, tone and the words we use, so that the audience understands the message we are trying to send. In a similar manner, we also have to be aware of the target audience when writing. Even though the task of adapting texts to different audiences may look easy to experienced writers, rookies and amateurs may struggle in selecting the information that might be relevant to a particular target audience. Nevertheless, a way to deal with words and some common sense should be enough to complete the task, but due to the latter requirement automating this task becomes a much harder problem.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

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

© 2020 Copyright held by the owner/author(s).

In this paper, we adapt texts to context by manipulating three text evaluation metrics: length, polarity and readability. Our method will be able to transition between social media posts, research articles, newspaper articles and official statements, where each publication type targets a different audience. While governmental institutions and academics both publish neutrally-oriented texts, research articles tend to be much longer than official statements. Social media and news usually target wider audiences, which is why texts should be more readable. However, the two can be separated by the amount of opinion we can include. Newspaper articles should be less biased and thus include less positively or negatively-oriented words.

Our methodology iteratively adapts key text evaluation metrics towards the mean values of the target publication type that will be calculated from a sample set of articles. In each iteration our method first manipulates length using abstractive summarization techniques and natural language generation models. Next, it replaces words with more appropriate synonyms and adjusts polarity and readability scores. Finally, it uses a fine-tuned text-to-text transformer to generate more appropriate paraphrases that replace whole sentences in our text and alter readability.

2 RELATED WORK

While we are trying to automatically adapt texts to a particular genre, researchers have already made progress in automatic text simplification, where we try to adapt text to be more readable and easier to understand. Carroll and Tait [2] developed a methodology to simplify texts for people that suffer from aphasia, which is a disability of language processing. The developed system consists of an analyser component, which provides syntactic analysis and a simplifier component, which adapts texts using lexical and syntactic simplification. Lexical simplifier replaces the words in text with synonyms by considering Kucera-Francis frequency of each available synonym that is held in WordNet. Syntactic constructions that are not constructed in Subject-Verb-Object order can also be tough to process for aphasic people. Therefore, the authors proposed several syntactic simplifications, such as replacement of passive constructions with active constructions.

A lot of research has already been done on how to evaluate and alter text and we will use many existing methods to help us develop our methodology. We picked three text evaluation metrics that can be reasonably altered using existing methods. Flesch [5] developed an equation that determines the readability of the text using the number of words per sentence and the number of syllables per word ratios. Even though structure-based metrics are important, we also have to consider the message of the text. Using sentiment analysis, we can determine whether the writer has positive or negative affections towards the topic of the text. Feldman [4] in his article discusses several approaches of sentiment analysis based on the unit that we will be classifying (i.e. documents, sentences, aspects).

As length is one of the chosen text evaluation metrics that we wish to adapt, we have to be able to both summarize and

extend the text. According to Allahyari et al. [1], we differentiate between extractive and abstractive summarization approaches. Extractive approaches shorten the original text by excluding less relevant sentences. Significance of the sentence can be evaluated by determining whether the sentence is related with the main topic or whether its content is distinctive in comparison to other sentences. On the other hand, abstractive approaches tend to summarize texts in a new (more human-like) manner by structuring the text into some logical form such as graphs, trees and ontologies [6].

When adapting shorter texts to longer, natural language generation has proven to be a very strong tool. Radford et al. [7] developed a natural language generation technique to generate additional text and produced state of the art results using unsupervised multitask learners for model learning. Their model was trained to predict the next word in text based on 40GB of Internet content. They concluded that large training datasets and models trained to maximize the likelihood of a sufficiently varied corpus can learn a surprising amount of tasks, while no supervision is needed in training.

Another method that is commonly used when adapting texts to context is paraphrasing, i.e. rewording of something written by changing its structure or replacing the words with their synonyms. Goutham in his article [9] used a pre-trained text-to-text transfer transformer to generate paraphrases of questions. The model was fine-tuned, where the input texts were questions from Quora and the expected output were the questions that were labeled as their duplicates.

In our paper, we plan to exploit the aforementioned abstractive summarization technique to shorten our texts and fine-tune the pre-trained natural language generation model that Radford et al. [7] developed. Similarly as Goutham [9], we intend to fine-tune a pre-trained text-to-text transformer that would be able to generate paraphrases of a sentence. To calculate readability score of the input text, we plan to use the formula proposed by Flesch [5].

3 ADAPTATION OF TEXT

As mentioned before, the proposed method iteratively manipulates the chosen text evaluation metrics to adapt text to different target audiences. In Figure 1, which gives an overview of the method, we can see that before we start running the process, we calculate the initial values of text evaluation metrics for each publication type as the average values of a set of documents. Our main dataset consists of 150 documents for each publication type, where all the documents hold text that contain COVID-19 related content, with which we minimize the effect of variables that we will not take into account in text adaptation. We also define the number of iterations (in our case: 5) and the acceptable error ϵ (in our case: $\epsilon = 0.1$) that determines whether it is still worth altering a particular text evaluation metric.

In each iteration, relative differences between current and initial values of text evaluation metrics are calculated. If the absolute relative difference to some metric is bigger than ϵ , we try to adjust it to the targeted value. We adjust key text evaluation metrics in the main loop of the process in Figure 1 using the following procedures:

- In case the target length is smaller than its current value, we use a pre-trained **T5 text-to-text transformer** [8] to **summarize** the input text. The model is an encoder-decoder model that uses transfer learning on a model that is

firstly pre-trained on a data-rich task using texts from the Colossal Clean Crawled Corpus and then fine-tuned on a downstream task using a dataset of texts and their summaries as the expected outputs from the aforementioned corpus.

- To generate additional text, if the input text is shorter than the average text of the target publication type, we use **fine-tuned natural language generation models**. We generate four pre-trained GPT2 natural language generation models [7] that are based on the aforementioned unsupervised multitask learners. Each model is then fine-tuned on a dataset of 100 texts of a certain considered publication type and should be able to generate texts similar to the ones that it was fine-tuned on. Consequently, we would assume that the generated text needs less further adaptation.
- While adapting length might be the procedure with the most visible results, we also have to adapt the other text evaluation metrics. We develop a **synonym replacement** procedure to adjust polarity and readability scores to the target values. The procedure is executed in iterations and in each iteration we replace the word with the highest sum of absolute relative differences of polarity and readability scores to the initial values of the target publication type with its optimal synonym, i.e. the synonym which causes the sum of absolute relative differences to minimize. We used the lexical database WordNet to acquire synonyms of the considered word.
- Finally, we alter readability by **generating paraphrases with a T5 text-to-text transformer** [9] that was fine-tuned to generate paraphrases by learning on Microsoft Research Paraphrase Corpus dataset [3]. We then pick the optimal paraphrase, which minimizes the relative difference to the target readability score.

Replacing sentences with their paraphrases could potentially also alter length and polarity. We test the assumption by generating five paraphrases for each sentence in 100 documents for each considered publication type and find that the relative difference of length and polarity between the initial sentence and its paraphrases is not significant. The obtained mean relative difference of polarity scores in this preliminary analysis was $0.91 \cdot 10^{-3}$ and the mean relative difference of lengths was $0.11 \cdot 10^{-3}$.

4 EVALUATION AND RESULTS

In our experiments, we evaluate the quality of text transformation between all possible pairs of four different publication media types: social media, news, research articles and official statements. We tested our methodology by generating adapted texts of a subset of the main dataset that was introduced in Section 3. The subset consists of 100 documents for each publication type (i.e., 400 altogether) that were randomly chosen from the main dataset. We adapted each document to the other three publication types and thus test all of the 12 possible transitions. We observed how the key text evaluation metrics behaved and whether the generated text was meaningful or not. The results text evaluation metrics before and after adaptation to context are shown in Table 1. In Table 2, we present the results of content quality evaluation of the generated texts.

From Table 1 we can observe that the text evaluation metrics successfully changed in the right direction. In most cases we

Input publication type		Target publication type							
		Official statements		Research articles		News		Social media	
		Initial	Adapted	Initial	Adapted	Initial	Adapted	Initial	Adapted
Official statements	Length			0.79	0.04	0.04	0.03	36.39	0.35
	Polarity			2.88	0.15	2.05	0.04	2.78	0.4
	Readability			0.36	0.75	0.23	0.35	0.4	0.24
Research articles	Length	3.06	0.05			2.99	0.04	136.23	0.33
	Polarity	0.81	0.27			0.33	0.07	0.18	0.46
	Readability	0.17	0.08			0.34	0.22	0.45	0.12
News	Length	0.97	0.03	0.99	0.03			63.79	0.4
	Polarity	0.88	0.14	0.43	0.1			0.33	0.37
	Readability	1.21	0.05	1.2	0.84			0.24	0.11
Social media	Length	0.69	0.02	0.64	0.03	0.97	0.04		
	Polarity	0.85	0.28	0.28	0.02	0.55	0.06		
	Readability	0.71	0.27	0.69	0.8	0.24	0.28		

Table 1: Absolute relative differences to initial values of target publication type before and after transition

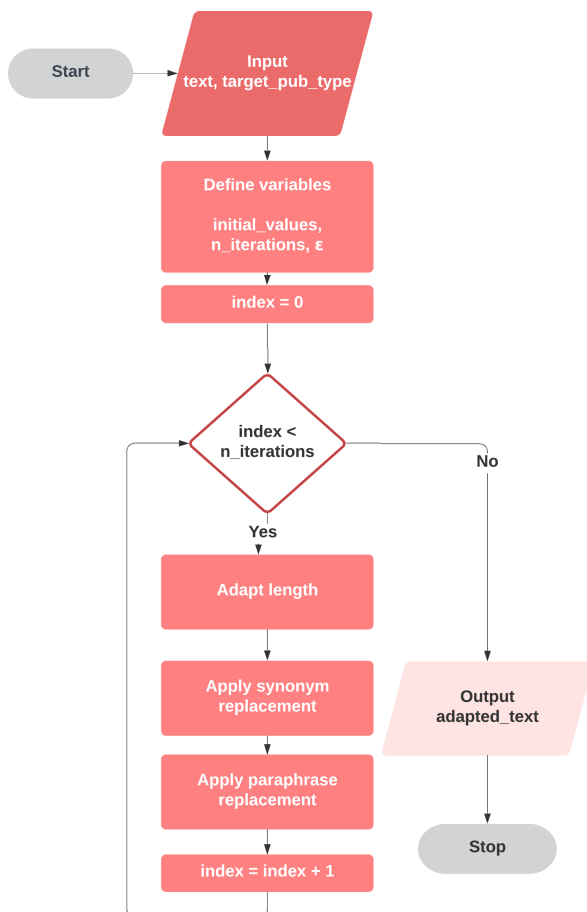


Figure 1: Flowchart of the text adaptation methodology

significantly improved the values of metrics. The length manipulation managed to consistently decrease the relative difference towards targeted length and in many occasions even converge under ϵ value. Polarity and readability scores seem harder to adapt. However, in each case we successfully adapted the sum of relative differences of those metrics, with which we can conclude

that synonym replacement method performs suitably, too. Its inefficiency may be caused by the lack of choice in synonym and paraphrase replacement and the limited amount of words and sentences that can be replaced.

As an example, we tried to adapt this research article to a social media post. By including statements that are colored in yellow in Figure 2, such as “The authors have proposed” and “The researchers used”, we imply that the social media post talks about a research article, which it does. Furthermore, the replacement of the word “texts” with “written matters” and the word “audiences” with “audience groups” indicates that the initial readability of this research article is higher than the expected value of social media posts, because we lower the Flesch Reading Ease score with the mentioned transformations. The content is appropriate as it extracts some of the most crucial concepts of this article.

The authors have proposed a way to adapt written matters to target publication types. The researchers used length, readability and polarity, all of which can be used to determine whether the text is suitable for different audience groups.

Figure 2: Example of text adaptation from this research article to a social media post

Additionally, we evaluated the content quality by checking semantic similarity between the input and the generated text. Using GloVe word embeddings, we transformed the text into vectors and calculated the angle between the vectors. With cosine measure we evaluated whether the vectors point in a similar direction, i.e. the contents of texts, are similar. In Table 2, we present the mean cosine similarities between GloVe embeddings of the input and the adapted texts. The results show that the generated texts preserve the original content. Cosine similarity scores are high in all transitions, however, the scores are a bit lower when we adapt to or from a social media post. This could be a consequence of the inability to thoroughly define the content in short texts that are expected in social media.

While our method successfully adapts key text evaluation metrics, our results are not perfect when it comes to the content. Our method has its drawbacks such as generating lots of additional content, which often results in an unconnected text. Additionally, synonym replacement and paraphrase generation

Original publication type \ Target publication type	Research article	Official statement	Social media	News
Research article		0.94	0.82	0.97
Official statement	0.95		0.82	0.97
Social media	0.83	0.93		0.90
News	0.95	0.96	0.82	

Table 2: Cosine similarities between GloVe embeddings

can incorrectly replace original sentence or word, where the paraphrase or synonym changes the meaning but proves to be efficient when adapting text evaluation metrics, if there exist such synonyms that are more appropriate to use for a particular target audience. Nevertheless, our methodology generated a few sequences that could be published for target audiences without any changes and lots of texts would only require minor corrections.

To conclude this section, we are satisfied with the benchmarking results that our method produced in adapting key text evaluation metrics. The methodology produces some interesting content and can thus be used as a baseline for further text adaptation to target audiences.

5 CONCLUSION

In this article we developed a methodology that adapts texts to context. The methodology focuses on three text evaluation metrics: length, readability and polarity of the text. Our method iteratively adapts text to the calculated initial values based on the targeted publication type by adjusting the key text evaluation metrics. We successfully managed to adjust text evaluation metrics in nearly all transitions.

While we found text evaluation metrics that define different publication types, in some cases adjusting these measures is not enough. Generating longer sequences of additional text, we find that the generated content is not connected and while we can find a chain of related topics of subsections, in some cases it is hard to define the common thread that is held throughout the whole text. Additionally, if such synonyms and paraphrases exist that corrupt the content but improve the relative differences to the targeted values of key text evaluation metrics, the methodology will replace existing words and sentences with senseless content. Despite these drawbacks, we generated lots of results that reflect the targeted publication types and even more results that would require only minor changes to be completely acceptable. We conclude this article with satisfactory results of both content of generated texts and their values of key text evaluation metrics.

Our ideas for further work include improvement of natural language generation model, where the pre-trained model that we used should be trained on longer texts so that we could generate text based on longer prompts and thus make sure that we hold the common thread throughout the whole text. Determining whether synonyms or paraphrases corrupt the message of the text is also very important. Word embeddings can be used to represent the context of the text and we could use it to determine whether the synonym fits the current context or not. Another way to adapt text to context would be to create a dataset of texts, where each row hold different versions of the same text and each version represents the text written for different target audience. This way we would be able to teach text-to-text models to adapt text to

context and the methodology could also consider patterns that might not be obvious to human's eye.

REFERENCES

- [1] Mehdi Allahyari, Seyedamin Pouriyeh, Mehdi Assefi, Saeid Safaei, Elizabeth Trippe, Juan Gutierrez, and Krys Kochut. 2017. Text summarization techniques: a brief survey. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 8, (July 2017), 397–405. doi: 10.14569/IJACSA.2017.081052.
- [2] John Carroll, Guido Minnen, Yvonne Canning, Siobhan Devlin, and John Tait. 1998. Practical simplification of english newspaper text to assist aphasic readers. *Proc. of AAAI-98 Workshop on Integrating Artificial Intelligence and Assistive Technology*, (July 1998), 7–10.
- [3] William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*, 9–16. <https://www.aclweb.org/anthology/I05-5002>.
- [4] Ronen Feldman. 2013. Techniques and applications for sentiment analysis. *Commun. ACM*, 56, (April 2013), 82–89. doi: 10.1145/2436256.2436274.
- [5] Rudolf Flesch. 1979. *How to Write Plain English: A Book for Lawyers and Consumers*. Harper & Row.
- [6] Kavita Ganesan, ChengXiang Zhai, and Jiawei Han. 2010. Opinosis: a graph based approach to abstractive summarization of highly redundant opinions. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*. Coling 2010 Organizing Committee, Beijing, China, (August 2010), 340–348. <https://www.aclweb.org/anthology/C10-1039>.
- [7] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2018. Language models are unsupervised multitask learners. <https://d4mucfpkisywv.cloudfront.net/better-language-models/language-models.pdf>.
- [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] Goutham Ramsri. 2020. Paraphrase any question with T5 (Text-To-Text Transfer Transformer). *Towards Data Science*. [Accessed: 17. 8. 2020]. (2020). <https://towardsdatascience.com/paraphrase-any-question-with-t5-text-to-text-transfer-transformer-pretrained-model-and-cbb9e35f1555>.

Indeks avtorjev / Author index

Adams Jennifer.....	104
Andova Andrejaana.....	7
Bierhoff Ilse.....	80
Bizjak Jani.....	27, 35, 39, 43, 51
Bizjak Miha.....	11
Bohanec Marko.....	27
Bolliger Larissa.....	63
Bosnić Zoran.....	76, 88, 100, 116
Brence Jure.....	104
Bromuri Stefano.....	7
Budna Borut.....	112
Clays Els.....	63
De Boer Jasmijn.....	80
De Masi Carlo M.....	15
Dolanc Gregor.....	27
Dovgan Erik.....	19, 92
Džeroski Sašo.....	104
Filipič Bogdan.....	19
Gams Matjaž.....	27, 35, 39, 43, 47, 51, 55, 68
Gazvoda Samo.....	35, 43
Gjoreski Hristijan.....	47, 72, 84
Gjoreski Martin.....	23
Golob David.....	27
Gradišek Anton.....	32
Guid Matej.....	11
Gültekin Várkonyi Gizem.....	32
Jordan Marko.....	80
Kalabakov Stefan.....	27, 35, 51
Katrašnik Marko.....	63
Kiprijanovska Ivana.....	39, 43, 47
Kocuvan Primož.....	27, 35, 51
Kolenik Tine.....	55
Kuzmanovski Vladimir.....	23
Levstek Andraž.....	59
Lukan Junoš.....	63
Luštrek Mitja.....	7, 15, 63, 80, 92, 96
Machidon Alina.....	68
Malina Edward.....	104
Mlakar Miha.....	112
Neceva Marija.....	72
Osipov Evgeny.....	76, 100
Peterka Ana.....	76
Petrovčič Janko.....	27
Ravničan Jože.....	27
Reščič Nina.....	80
Shulajkavska Miljana.....	84
Silan Darja.....	59
Simončič Žiga.....	88
Slapničar Gašper.....	92
Smerkol Maj.....	68, 112
Stankoski Simon.....	96
Štepec Dejan.....	108
Stoilkovska Emilija.....	72
Stropnik Vid.....	100
Szlupowicz Michał Artur.....	104

Valič Jakob.....	92
Vodopija Aljoša	59
Žontar Luka	116
Zupančič Jernej	112

IS
20
20

Slovenska konferenca o umetni inteligenci
Slovenian Conference on Artificial Intelligence

Mitja Luštrek, Matjaž Gams, Rok Piltaver