Zbornik 20. mednarodne multikonference

INFORMACIJSKA DRUŽBA - IS 2017

Zvezek A

Proceedings of the 20th International Multiconference

INFORMATION SOCIETY - IS 2017

Volume A

Slovenska konferenca o umetni inteligenci
Slovenian Conference on Artificial Intelligence

Uredili / Edited by
Matjaž Gams, Mitja Luštrek, Rok Piltaver

http://is.ijs.si

9.–13. oktober 2017 / 9–13 October 2017
Ljubljana, Slovenia

Zbornik 20. mednarodne multikonference

INFORMACIJSKA DRUŽBA – IS 2017

Zvezek A

Proceedings of the 20th International Multiconference

INFORMATION SOCIETY – IS 2017

Volume A

Slovenska konferenca o umetni inteligenci
Slovenian Conference on Artificial Intelligence

Uredili / Edited by

Mitja Luštrek, Rok Piltaver, Matjaž Gams

http://is.ijs.si

9. - 13. oktober 2017 / 9th – 13th October 2017
Ljubljana, Slovenia

Uredniki:

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

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

Matjaž Gams
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 2017

Kataložni zapis o publikaciji (CIP) pripravili v Narodni in univerzitetni knjižnici v Ljubljani
COBISS.SI-ID=1537600707
ISBN 978-961-264-112-2 (pdf)

PREDGOVOR MULTIKONFERENCI INFORMACIJSKA DRUŽBA 2017

Multikonferenca Informacijska družba (http://is.ijs.si) je z **dvajseto** zaporedno prireditvijo osrednji srednjeevropski dogodek na področju informacijske družbe, računalništva in informatike. Letošnja prireditev je ponovno na več lokacijah, osrednji dogodki pa so na Institutu »Jožef Stefan«.

Informacijska družba, znanje in umetna inteligenca so spet na razpotju tako same zase kot glede vpliva na človeški razvoj. Se bo eksponentna rast elektronike po Moorovem zakonu nadaljevala ali stagnirala? Bo umetna inteligenca nadaljevala svoj neverjetni razvoj in premagovala ljudi na čedalje več področjih in s tem omogočila razcvet civilizacije, ali pa bo eksponentna rast prebivalstva zlasti v Afriki povzročila zadušitev rasti? Čedalje več pokazateljev kaže v oba ekstrema – da prehajamo v naslednje civilizacijsko obdobje, hkrati pa so planetarni konflikti sodobne družbe čedalje težje obvladljivi.

Letos smo v multikonferenco povezali dvanajst odličnih neodvisnih konferenc. Predstavljenih bo okoli 200 predstavitev, povzetkov in referatov v okviru samostojnih konferenc in delavnic. Prireditev bodo spremljale okrogle mize in razprave ter posebni dogodki, kot je svečana podelitev nagrad. Izbrani prispevki bodo izšli tudi v posebni številki revije Informatica, ki se ponaša s **40-letno** tradicijo odlične znanstvene revije. Odlične obletnice!

Multikonferenco Informacijska družba 2017 sestavljajo naslednje samostojne konference:

- Slovenska konferenca o umetni inteligenci
- Soočanje z demografskimi izzivi
- Kognitivna znanost
- Sodelovanje, programska oprema in storitve v informacijski družbi
- Izkopavanje znanja in podatkovna skladišča
- Vzgoja in izobraževanje v informacijski družbi
- Četrta študentska računalniška konferenca
- Delavnica »EM-zdravje«
- Peta mednarodna konferenca kognitonike
- Mednarodna konferenca za prenos tehnologij - ITTC
- Delavnica »AS-IT-IC«
- Robotika

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 inštitucijam, 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 2017 bomo petič podelili nagrado za življenjske dosežke v čast Donalda Michija in Alana Turinga. Nagrado Michie-Turing za izjemen življenjski prispevek k razvoju in promociji informacijske družbe bo prejel prof. dr. Marjan Krisper. Priznanje za dosežek leta bo pripadlo prof. dr. Andreju Brodniku. Že šestič podeljujemo nagradi »informacijska limona« in »informacijska jagoda« za najbolj (ne)uspešne poteze v zvezi z informacijsko družbo. Limono je dobilo padanje slovenskih sredstev za akademsko znanost, tako da smo sedaj tretji najslabši po tem kriteriju v Evropi, jagodo pa »e-recept«. Čestitke nagrajencem!

Bojan Orel, predsednik programskega odbora
Matjaž Gams, predsednik organizacijskega odbora

FOREWORD - INFORMATION SOCIETY 2017

In its **20th year**, the Information Society Multiconference (http://is.ijs.si) remains one of the leading conferences in Central Europe devoted to information society, computer science and informatics. In 2017 it is organized at various locations, with the main events at the Jožef Stefan Institute.

The pace of progress of information society, knowledge and artificial intelligence is speeding up, and it seems we are again at a turning point. Will the progress of electronics continue according to the Moore's law or will it start stagnating? Will AI continue to outperform humans at more and more activities and in this way enable the predicted unseen human progress, or will the growth of human population in particular in Africa cause global decline? Both extremes seem more and more likely – fantastic human progress and planetary decline caused by humans destroying our environment and each other.

The Multiconference is running in parallel sessions with 200 presentations of scientific papers at twelve conferences, round tables, workshops and award ceremonies. Selected papers will be published in the Informatica journal, which has **40 years** of tradition of excellent research publication. These are remarkable achievements.

The Information Society 2017 Multiconference consists of the following conferences:

- Slovenian Conference on Artificial Intelligence
- Facing Demographic Challenges
- Cognitive Science
- Collaboration, Software and Services in Information Society
- Data Mining and Data Warehouses
- Education in Information Society
- 4th Student Computer Science Research Conference
- Workshop Electronic and Mobile Health
- 5th International Conference on Cognitonics
- International Conference of Transfer of Technologies - ITTC
- Workshop »AC-IT-IC«
- Robotics

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 fifth year, the award for life-long outstanding contributions will be delivered in memory of Donald Michie and Alan Turing. The Michie-Turing award will be given to Prof. Marjan Krisper for his life-long outstanding contribution to the development and promotion of information society in our country. In addition, an award for current achievements will be given to Prof. Andrej Brodnik. The information lemon goes to national funding of the academic science, which degrades Slovenia to the third worst position in Europe. The information strawberry is awarded for the medical e-recipe project. Congratulations!

Bojan Orel, 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
Karl Pribram, 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
Toby Walsh, Australia

Organizing Committee

Matjaž Gams, chair
Mitja Luštrek
Lana Zemljak
Vesna Koricki
Mitja Lasič
Robert Blatnik
Aleš Tavčar
Blaž Mahnič
Jure Šorn
Mario Konecki

Programme Committee

Bojan Orel, chair
Franc Solina, co-chair
Viljan Mahnič, co-chair
Cene Bavec, co-chair
Tomaž Kalin, co-chair
Jozsef Györkös, co-chair
Tadej Bajd
Jaroslav Berce
Mojca Bernik
Marko Bohanec
Ivan Bratko
Andrej Brodnik
Dušan Caf
Saša Divjak
Tomaž Erjavec
Bogdan Filipič
Andrej Gams
Matjaž Gams
Mitja Luštrek
Marko Grobelnik
Nikola Guid
Marjan Heričko
Borka Jerman Blažič Džonova
Gorazd Kandus
Urban Kordeš
Marjan Krisper
Andrej Kuščer
Jadran Lenarčič
Borut Likar
Janez Malačič
Olga Markič
Dunja Mladenič
Franc Novak
Vladislav Rajkovič
Grega Repovš
Ivan Rozman
Niko Schlamberger
Stanko Strmčnik
Jurij Šilc
Jurij Tasič
Denis Trček
Andrej Ule
Tanja Urbančič
Boštjan Vilfan
Baldomir Zajc
Blaž Zupan
Boris Žemva
Leon Žlajpah

Invited lecture

AN UPDATE FROM THE AI & MUSIC FRONT

Gerhard Widmer
Institute for Computational Perception
Johannes Kepler University Linz (JKU), and
Austrian Research Institute for Artificial Intelligence (OFAI), Vienna

Abstract

Much of current research in Artificial Intelligence and Music, and particularly in the field of Music Information Retrieval (MIR), focuses on algorithms that interpret musical signals and recognize musically relevant objects and patterns at various levels -- from notes to beats and rhythm, to melodic and harmonic patterns and higher-level segment structure --, with the goal of supporting novel applications in the digital music world. This presentation will give the audience a glimpse of what musically "intelligent" systems can currently do with music, and what this is good for. However, we will also find that while some of these capabilities are quite impressive, they are still far from (and do not require) a deeper "understanding" of music. An ongoing project will be presented that aims to take AI & music research a bit closer to the "essence" of music, going beyond surface features and focusing on the expressive aspects of music, and how these are communicated in music. This raises a number of new research challenges for the field of AI and Music (discussed in much more detail in [Widmer, 2016]). As a first step, we will look at recent work on computational models of expressive music performance, and will show some examples of the state of the art (including the result of a recent musical 'Turing test').

References

Widmer, G. (2016).
Getting Closer to the Essence of Music: The Con Espressione Manifesto.
ACM Transactions on Intelligent Systems and Technology 8(2), Article 19.

KAZALO / TABLE OF CONTENTS

Zbornik 20. mednarodne multikonference

INFORMACIJSKA DRUŽBA – IS 2017

Zvezek A

Proceedings of the 20th International Multiconference

INFORMATION SOCIETY – IS 2017

Volume A

Slovenska konferenca o umetni inteligenci
Slovenian Conference on Artificial Intelligence

Uredili / Edited by

Mitja Luštrek, Rok Piltaver, Matjaž Gams

http://is.ijs.si

12. - 13. oktober 2017 / 12th – 13th October 2017
Ljubljana, Slovenia

PREDGOVOR

V letu 2017 smo bili spet priča neverjetnim dosežkom umetne inteligence, ki na čedalje več področjih prekaša človeške sposobnosti. Velja omeniti poker Texas hold'em brez omejitev pri višini stav (ki je precej bolj kompleksen od že rešene različice z omejitvami) in strateško računalniško igro Dota 2, kjer so se do sedaj ljudje uspešno upirali programom umetne inteligence, sedaj pa je v igri ena na ena umetna inteligenca pokazala premoč. Podobno dobro rešuje tudi resnejše probleme, npr. prepoznavanje rakavih tkiv za zgodnjo diagnozo, kjer pa opažamo počasen prenos dosežkov iz raziskovalnih laboratorijev v prakso. Umetna inteligenca že sedaj ljudem pomaga na veliko področjih in celo rešuje življenja. Trendi kažejo, da bo naslednje leto še bolj koristna in prijazna. In naslednja leta še bolj.

Mnoge zanimive dosežke umetne inteligence lahko spoznamo tudi na Slovenski konferenci o umetni inteligenci (SKUI). Letos smo sprejeli 21 prispevkov, kar so trije več kot lani. Kot pretekla leta jih je največ z Instituta »Jožef Stefan«. Obžalujemo, da jih je manj kot lani prispevala Fakultete za računalništvo in informatiko, ki ima skupaj z Institutom vodilno vlogo pri raziskavah umetne inteligence v Sloveniji, pozdravljamo pa dva zelo kakovostna prispevka iz industrije. Upamo, da bo prispevkov iz industrije in nasploh izven Instituta prihodnja leta še več, saj je ključen cilj SKUI povezovanje vseh slovenskih raziskovalcev umetne inteligence, čeprav na konferenci niso nič manj dobrodošli tudi prispevki iz drugih držav.

SKUI je naslednica konference Inteligentni sistemi, ki je sestavni del multikonference Informacijska družba že od njenega začetka leta 1997. Letos tako skupaj s celotno multikonferenco praznuje 20. obletnico. Ker poleg tega Slovensko društvo za umetno inteligenco (SLAIS) – ki SKUI šteje za svojo konferenco – praznuje 25. obletnico, smo se odločili razširjene različice najboljših prispevkov povabiti v posebno številko revije Informatica o umetni inteligenci. Objava najboljših prispevkov z Informacijske družbe v Informatici je že dolga tradicija, ki pa jo bomo letos s posebno številko revije, kjer bomo objavili izbrane raziskave umetne inteligence v Sloveniji, še oplemenitili.

Mitja Luštrek, Rok Piltaver, Matjaž Gams

FOREWORD

2017 has brought many exciting achievements of artificial intelligence, which is proving superior to humans in increasingly many fields. Two examples are no-limit Texas hold'em poker (which is substantially more complex than the already solved limit version) and the computer strategy game Dota 2. In both cases, artificial intelligence has not been able to match the best humans so far, but this changed this year. Artificial intelligence is also solving more serious problems, such as the identification of cancerous tissue to enable early diagnosis; unfortunately, though, such achievements are not translated from research laboratories to practice as quickly as we may wish. However, artificial intelligence is already helping people in many fields and even saving lives. Trends indicate that it will be even more useful and friendly next year, and more so the years after that.

Slovenian Conference on Artificial Intelligence (SCAI) is a venue where one can learn about many achievements of artificial intelligence. 21 papers were accepted this year, which is three more than previous year. As in past years, most of them were from Jožef Stefan Institute. We regret that the Faculty of Computer and Information Science, which shares the leading role in artificial intelligence research in Slovenia with the Institute, contributed fewer papers this year; however, we are glad to have received two very high-quality papers from the industry. We hope for even more papers from the industry and other institutions outside the Institute in the following years, since a key objective of the conference is bringing together all Slovenian artificial intelligence researchers, although international papers are of course equally welcome.

SCAI is the successor of the Intelligent Systems conference, which has been a part of the Information Society multiconference since its establishment in 1997. The conference – together with the whole multiconference – thus celebrates its 20th anniversary this year. In addition, Slovenian Artificial Intelligence Society (SLAIS), which is the main supporter of SCAI, celebrates its 25th anniversary. Because of that, the extended versions of the best papers will be invited to a special issue of the Informatica journal on artificial intelligence. Publishing the best papers from the Information Society conference in the Informatica journal has a long tradition, but this year the best SCAI papers will find themselves in the company of other selected papers on the Slovenian research on artificial intelligence.

Mitja Luštrek, Rok Piltaver, Matjaž Gams

PROGRAMSKI ODBOR / PROGRAMME COMMITTEE

Mitja Luštrek, IJS (co-chair)

Rok Piltaver, IJS (co-chair)

Matjaž Gams, IJS (co-chair)

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

Artificial Intelligence in 2017

Matjaž Gams
Jozef Stefan Institute
Jamova 39, 1000 Ljubljana, Slovenia
matjaz.gams@ijs.si

ABSTRACT

In recent years, AI is facing incredibly fast progress. In this paper we review a couple of major new AI-related achievements and events. Among them, **IJCAI 2017** as the cover AI worldwide conference presented major scientific and industrial achievements along with several discussions and panels. Among them was AI superiority in the unlimited Texas hold'em poker and Dota 2. Both games were slightly limited, e.g. only 2 players instead of 10 in Dota 2, but the games itself included all major components such as bluffing with hidden cards or properties of dynamic strategic game with global and local decision making. Deep Neural Networks continue their excellence in **visual recognition tasks** and in real-life diagnostics, such as diagnosing which tissue contains malignant cancer cells, exceed best human experts in more and more diseases.

Among broader influence of AI on human future life, the **ban of autonomous weapons** was steadily promoted and as a result, the **asilomar principles** were defined for the first time. The principles present an attempt to provide guidelines for human-beneficial AI, the one that would prevent possibilities for AI to turn into human-harmful ways. The aim of the paper is to bring these issues to our society through presentation and discussions.

Keywords

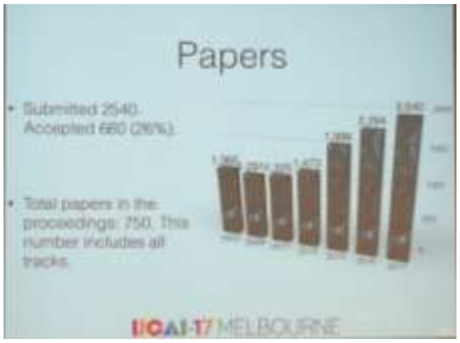
Artificial intelligence, AI principles, Future of life institute

1. INTRODUCTION

The progress of artificial intelligence (AI) is fast and often surprisingly efficient even for AI professionals [5]. Each year there are scores of new achievements in academia, gaming, industry, and real life. There are also practical modifications of the way we live and work. For example, autonomous vehicles are improving constantly, they are introduced into more and more countries. In Europe, the goal to introduce similar legislation promoting the drones and autonomous vehicles alike by the EU Commissioner Violeta Bulc has not been successful yet, while several EU countries have modified their traffic laws accordingly and USA has recently changed its legislation to promote faster implementation of autonomous vehicle into real life. In Slovenia, where the Justice Minister Goran Klemenčič is intensively trying to modernize the legal system despite the resistance of mainly status-quo majority, the drones are prohibited to spam the space, but as it is becoming a European habit, the bureaucratic viewpoint prohibits the use of drones and autonomous vehicles also for scientific purposes. As a result, Slovenian researchers are developing drones and autonomous cars illegally, but luckily nobody charges them for that. This is just one example how the political and legal system is lagging behind the progress of artificial intelligence and ICT – information and communication technologies.

2. IJCAI 2017

The 26th International Joint Conference on Artificial Intelligence was held in Melbourne, Australia in August 2017 [6]. Melbourne is world's most liveable city for seventh year running and indeed it is safe, clean, not crowded, full of green nature and architectural wonders.

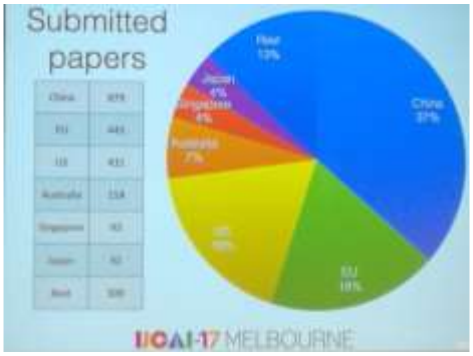


Papers

- Submitted 2540. Accepted 660 (26%).
- Total papers in the proceedings: 750. This number includes all tracks.

Figure 1: The growth of IJCAI papers in recent years.

The AI growth is indicated by the number of papers submitted to the IJCAI conference (Figure 1). In 2016 in New York there were 2.294 papers submitted while in 2017 in Melbourne, 2540 papers were reviewed. The growth was steady from 2009 on.



Submitted papers

Figure 2: Papers per countries at IJCAI 2017.

Study of papers submitted per country (Figure 2) at IJCAI 2017 indicates that the majority of them was from China (37%), second EU (18%) and third US (18%).

On September 1, Vladimir Putin speaking with students warned that whoever cracks artificial intelligence will 'rule the world' [9]. Will that be China since it already submits the major bulk of AI papers? Or will it be USA since most of the awards were given to USA researchers?

It is not only the number of AI papers from China, the industry achievements are astonishing as well. One might not be as familiar with the Chinese solutions as with Google or Amazon AI systems, but Chinese systems are close to the top. For example, in 2017 China's Alibaba Group Holding Ltd introduced a cut-price voice assistant speaker, similar to Amazon.com Inc's "Echo". It is named "Tmall Genie" and costs $73, significantly less than western counterparts by Amazon and Alphabet Inc's Google, which range around $150. Similarly, Baidu, China's top search engine, recently launched a device based on its own Siri-like "Duer OS" system. Alibaba and China's top tech firms have ambitions to become world leaders in artificial intelligence as companies.

In terms of overall several scientific and practical achievements presented at IJCAI 2017, two games stood out as another example of AI beating the best human counterparts: unlimited Texas hold'em poker (10 on 160 possibilities) and Dota 2. Both games were slightly limited - in poker, there are only two players, and Dota 2 was also reduced to only two players instead of 10. Nevertheless, both games are most-played human games with award funds going into tens of millions. Both games are quite different from formal games like chess or Go. For example, poker included all major components of human bluffing interactions and hidden cards. Dota 2 was constructed in a way that fast computers had no advantage and the outcome of a game was dependent on strategic plans with global and local decision making, and adapting to the adversary. From Wikipedia: "Dota 2 is originally played in matches between two teams of five players, with each team occupying and defending their own separate base on the map. Each of the ten players independently controls a powerful character, known as a "hero", who all have unique abilities and differing styles of play. During a match, the player collects experience points and items for their heroes in order to successfully fight the opposing team's heroes, who are doing the same. A team wins by being the first to destroy a large structure located in the opposing team's base, called the "Ancient", which is guarded by defensive towers."

Regarding the methods, reinforcement learning and deep neural networks were somehow most common applied, however, the AI field was presented through over 10 major areas.

Deep Neural Networks continue their excellence in visual recognition tasks and in real-life diagnostics, such as diagnosing which tissue contains malignant cancer cells, exceed best human experts in more and more diseases. There are several tasks, e.g. recognition of faces from a picture where DNNs recognized hundreds of faces in seconds, a result no human can match. Figure 3 demonstrates the progress of DNNs in visual tasks: around 2015 the visual recognition in specific domains was comparable to humans. Now, it is surpassed humans quite significantly – again, in particular visual tests.

The effects of only visual superiority are astonishing on its own. For example, eye analyses enable detecting certain diseases like cancer or Alzheimer [3]. Furthermore, DNN studies of facial properties enable detecting sexual orientation, IQ, and political orientation. When shown five photos of each man, a recent system was able to correctly select the man's sexuality 91 per cent of the time, while humans were able to perform the same task with less than 70% accuracy [7]. This Stanford University study alone confirmed that homosexuality is very probably of genetic origin. The consequences of one single study can be profound. Will job applications be determined also by the DNN study of facial properties? Will dictatorship countries prosecuting homosexuality punish their citizens on the basis of their faces?

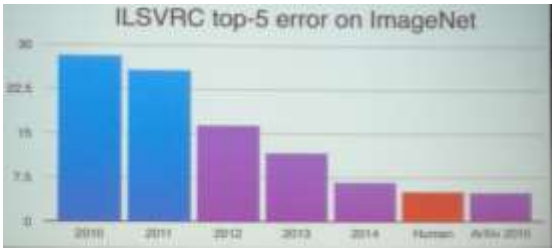


Figure 3: Error of DNNs on ImageNet through years.

There were several demonstrations and competitions at IJCAI 2017, including the traditional Angry birds competition. Most attractive, however, were soccer competitions with off-line Nao robots that were not trained or advised as a team, but performed on their own in a group proclaimed at the spot. Unfortunately, the local computing powers are at the level of a mobile phone, insufficient for good play. In Figure 4 one can see robots wondering around and searching for a ball. Still, they demonstrated some quite cunning properties, e.g. precise kicking the ball into the goal under the desired angle compared to the foot.

Figure 4: Soccer competition of independent individual Nao robots, dynamically assembled into teams at IJCAI 2017.

Next year will be of particular interest. ICML with 3500 attendees, IJCAI+ECAI with 2500, AAMAS with 700, ICCBR with 250 and SOCS with 50 attendees will be hosted at Stockholm in a 2-week event, July 2018. Wishful jokes are emerging that the critical mass of 6-7000 attendees will provide the critical mass to ignite the general intelligence or even superintelligence [2, 8, 10].

3. BAN OF AUTONOMOUS WEAPONS

Due to the vicinity of the Syrian conflict it is interesting to observe the level of sophistication of ICT solutions. ISIL, despite its technical inferiority, was the first to use slightly modified industrial drones to drop small bombs on the infantry. They also use remotely controlled weapons such as machineguns. However, the often used suicide industrial cars, fully loaded with explosives and shielded by attached armor plates, are still driven by vulnerable humans and not by remote controls on both sides.

None of these weapons falls into the category of fully autonomous weapons AI scientists propose to ban since they don't decide on its own when to fire.

There are two major reasons for the proposed ban:

- The fully autonomous weapons will likely make the war inhumane whereas humans – if war cannot be avoided – need some rule of engagement to preserve some level of humanity and prevent too extreme human suffering.

- This is one of preconditions on the road to prevent superintelligence to go viral, malignant [2, 8, 10].

There is some reason for celebrating the first successes of the pro-ban efforts – the movement is spreading through the social media since it started years ago by scientists like Toby Walsh or Stuart Russel and is currently coordinated by Mary Wareham. Slovenia is involved at national level where 4 societies (SLAIS for artificial intelligence, DKZ for cognitive science, Informatica for informatics, ACM Slovenia for computer science) assembled a letter and sent it to the UN and Slovenian government, while lately the Slovenian AI society SLAIS submitted a letter to the European national communities to join activities in this direction. Our initiative was also debated at the EurAI meeting at IJCAI 2017.

Second, Elon Mask and CEOs of 155 robotic companies assembled a letter, in which they write "Once developed, lethal autonomous weapons will permit armed conflict to be fought at a scale greater than ever, and at timescales faster than humans can comprehend. These can be weapons of terror, weapons that despots and terrorists use against innocent populations, and weapons hacked to behave in undesirable ways."

"We do not have long to act. Once this Pandora's box is opened, it will be hard to close."

On the other hand, the world superpowers are rapidly not only developing, but also applying autonomous weapons from drones to tanks or submarines. Some even argue that it is already too late to stop the autonomous weapons

Another example: the EU parliament accepted a new legislation giving artificial systems some rights of live beings. This is exactly one of the rules of the thumb not to do to avoid the potentially negative AI progress. So, why did the EU politicians accept such a law? It is not dangerous yet, but clearly worrisome.

4. The 23 ASILOMAR PRINCIPLES

The Future of Life Institute's [4] second conference on the future of artificial intelligence was organized in January 2017. The purpose of this paper is to present, in a rather original way as presented at the conference, the 23 asilomar AI principles [1] defined at the BAI 2017 conference, accompanied with the original discussions, the comments and analysis of the author of this paper.

The opinion of the community is pretty a shared one: "a major change is coming, over unknown timescales but across every segment of society, and the people playing a part in that transition have a huge responsibility and opportunity to shape it for the best."

The first task of the organizers was to compile a list of scores of opinions about what society should do to best manage AI in coming decades. From this list, the organizers distilled as much as they could into a core set of principles that expressed some level of consensus. The coordinating effort was dominating the event, resulting in a significantly revised version for use at the meeting. There, small breakout groups discussed subsets of the principles, giving detailed refinements and commentary on them. This process generated improved versions of the principles. Finally, they surveyed the full set of attendees to determine the level of support for each version of each principle.

After the consuming and meticulous process, a high level of consensus emerged around many of the statements during that final survey. The final list retained the principles if at least 90% of the attendees agreed on them. The 23 principles were grouped into research strategies, data rights and future issues including potential superintelligence, signed by those wishing to associate their name with the list. The principles will hopefully provide some guidelines as to how the power of AI can be used to improve everyone's lives in coming years.

At the web page of the event on the web pages of the Future of Life Institute [4], the following original presentations can be obtained with additional interviews on the consequent links

Artificial intelligence has already provided beneficial tools that are used every day by people around the world. Its continued development, guided by the following principles, will offer amazing opportunities to help and empower people in the decades and centuries ahead.

4.1 Research Issues

1) Research Goal: The goal of AI research should be to create not undirected intelligence, but beneficial intelligence.

2) **Research Funding:** Investments in AI should be accompanied by funding for research on ensuring its beneficial use, including thorny questions in computer science, economics, law, ethics, and social studies, such as:

- How can we make future AI systems highly robust, so that they do what we want without malfunctioning or getting hacked?
- How can we grow our prosperity through automation while maintaining people's resources and purpose?
- How can we update our legal systems to be more fair and efficient, to keep pace with AI, and to manage the risks associated with AI?
- What set of values should AI be aligned with, and what legal and ethical status should it have?

3) **Science-Policy Link:** There should be constructive and healthy exchange between AI researchers and policy-makers.

4) **Research Culture:** A culture of cooperation, trust, and transparency should be fostered among researchers and developers of AI.

5) **Race Avoidance:** Teams developing AI systems should actively cooperate to avoid corner-cutting on safety standards.

4.2 Ethics and Values

6) **Safety:** AI systems should be safe and secure throughout their operational lifetime, and verifiably so where applicable and feasible.

7) **Failure Transparency:** If an AI system causes harm, it should be possible to ascertain why.

8) **Judicial Transparency:** Any involvement by an autonomous system in judicial decision-making should provide a satisfactory explanation auditable by a competent human authority.

9) **Responsibility:** Designers and builders of advanced AI systems are stakeholders in the moral implications of their use, misuse, and actions, with a responsibility and opportunity to shape those implications.

10) **Value Alignment:** Highly autonomous AI systems should be designed so that their goals and behaviors can be assured to align with human values throughout their operation.

11) **Human Values:** AI systems should be designed and operated so as to be compatible with ideals of human dignity, rights, freedoms, and cultural diversity.

12) **Personal Privacy:** People should have the right to access, manage and control the data they generate, given AI systems' power to analyze and utilize that data.

13) **Liberty and Privacy:** The application of AI to personal data must not unreasonably curtail people's real or perceived liberty.

14) **Shared Benefit:** AI technologies should benefit and empower as many people as possible.

15) **Shared Prosperity:** The economic prosperity created by AI should be shared broadly, to benefit all of humanity.

16) **Human Control:** Humans should choose how and whether to delegate decisions to AI systems, to accomplish human-chosen objectives.

17) **Non-subversion:** The power conferred by control of highly advanced AI systems should respect and improve, rather than subvert, the social and civic processes on which the health of society depends.

18) **AI Arms Race:** An arms race in lethal autonomous weapons should be avoided.

4.3 Longer-term Issues

19) **Capability Caution:** There being no consensus, we should avoid strong assumptions regarding upper limits on future AI capabilities.

20) **Importance:** Advanced AI could represent a profound change in the history of life on Earth, and should be planned for and managed with commensurate care and resources.

21) **Risks:** Risks posed by AI systems, especially catastrophic or existential risks, must be subject to planning and mitigation efforts commensurate with their expected impact.

22) **Recursive Self-Improvement:** AI systems designed to recursively self-improve or self-replicate in a manner that could lead to rapidly increasing quality or quantity must be subject to strict safety and control measures.

23) **Common Good:** Superintelligence should only be developed in the service of widely shared ethical ideals, and for the benefit of all humanity rather than one state or organization.

5. CONCLUSION

The AI progress is already fascinating, and it speeds-up each consequent year. The rising awareness of AI-related changes in human society are appearing in scientific, academia and general public. Dozens of major reports have emerged from academia (e.g. the Stanford 100-year report), government (e.g. two major reports from the White House), industry (e.g. materials from the Partnership on AI), and the nonprofit sector (e.g. a major IEEE report). The paper will hopefully spur discussion and awareness about these issues also in our country where it is most important that the public, media and governance understand that the times are changing fast, that new approaches and methods are needed.

Scientific comprehensions about AI, its influence on everyday life, and future for the human civilization are stacking up. Scientists are able to provide some guidelines in which direction should we humans develop AI to avoid the dangers of the negative effects of the rising power of artificial intelligence. While AI often frightens general public, this author finds its fast progress a necessity to prevent degradation or self-destruction of human civilization. The potential dangers are real, not fictitious, primarily to a simple fact that any major power can be easily misused to cause harm to humans, and second, that there are some strong indications that civilizations tend to destroy themselves. By raising awareness, we increase the chances to ripe the positive aspects of the future mighty AI and avoid the negative ones.

6. REFERENCES

[1] Asilomar principles. 2017, (https://futureoflife.org/2017/01/17/principled-ai-discussion-asilomar/).

[2] Bostrom, N. 2014. *Superintelligence – Paths, Dangers, Strategies.* Oxford University Press, Oxford, UK.

[3] Eye Scans to Detect Cancer and Alzheimer's Disease, https://spectrum.ieee.org/the-human-os/biomedical/diagnostics/eye-scans-to-detect-cancer-and-alzheimers-disease

[4] Future of life institute, https://futureoflife.org/

[5] Gams, M. 2001. *Weak intelligence: through the principle and paradox of multiple knowledge*. Nova Science.

[6] IJCAI conference, 2017, https://ijcai-17.or

[7] Kosinski, M., Wang. Y. 2017. Deep neural networks are more accurate than humans at detecting sexual orientation from facial images. https://osf.io/zn79k/

[8] Kurzweil, R. 2006. T*he Singularity Is Near: When Humans Transcend Biology*, Sep 26, Penguin Books.

[9] Mail online, Science and technology, Vladimir Putin warns whoever cracks artificial intelligence will 'rule the world', http://www.dailymail.co.uk/sciencetech/article-4844322/Putin-Leader-artificial-intelligence-rule-world.html

[10] Yampolskiy, R.V. 2016. *Artificial Superintelligence*. CRC Press.

Comparison of Feature Ranking Approaches for Discovery of Rare Genetic Variants Related to Multiple Sclerosis

Matej Petković [1,2,✉]
matej.petkovic@ijs.si

Jovan Tanevski [2]

Aleš Maver [3]

Lovro Vidmar [3]

Borut Peterlin [3]

Sašo Džeroski [1,2]

[1] International Postgraduate School Jožef Stefan, Ljubljana, Slovenia
[2] Department of Knowledge Technologies, Jožef Stefan Institute, Ljubljana, Slovenia
[3] Clinical Institute of Medical Genetics, University Medical Centre Ljubljana, Slovenia

ABSTRACT

In this work, we assess the quality of the ReliefF and Genie3 feature ranking algorithms on the task of discovering rare genetic variants related to multiple sclerosis using real world data. The data consists of a total of 183 patients with multiple sclerosis and healthy controls. We evaluate the rankings and check whether two different environments for data acquisition influence the data. The results show the that Genie3 algorithm produces better rankings. However, different environments for data acquisition have lesser influence on the ReliefF rankings.

Categories and Subject Descriptors

I.5.2 [**Pattern Recognition**]: Design Methodology

General Terms

Algorithms, Experimentation

Keywords

feature ranking, genetic variants, multiple sclerosis

1. INTRODUCTION

Feature ranking (FR) is an important task in machine learning, which can be formalized as follows. We are given a set of examples $\boldsymbol{x}$ from the input domain $\mathcal{X} \subseteq \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_D$, where $D \geq 1$ is the number of descriptive attributes (features). We assume that the domain $\mathcal{X}_i$ of the i-th feature x_i is either a subset of $\mathbb{R}$ or an arbitrary finite set, i.e., domain $\mathcal{X}_i$ and feature x_i are either numeric or nominal. Each example $\boldsymbol{x}$ is associated with a target value $y(\boldsymbol{x})$ from the target domain $\mathcal{Y}$. Given a dataset $\mathscr{D} \subseteq \mathcal{X} \times \mathcal{Y}$, the goal of FR is to estimate how much each of the features influences the target, and then order the features with respect to the influences.

FR is a significant part of predictive modelling. The goal of predictive modelling is to learn a model able to predict the values of the target variable y, given a dataset $\mathscr{D}$. The two general types of predictive modelling are regression (when $\mathcal{Y} \subseteq \mathbb{R}$) and classification (otherwise). In our work we are concerned with classification. In classification, the values from $\mathcal{Y}$ are usually referred to as classes.

There are three main reasons for FR with regards to predictive modeling. First, we may want to reduce the dimensionality of the input space, so that only the features that contain the most information about target are kept in the dataset. By doing this, we decrease the amount of memory/time needed to build a predictive model, while the performance of the model is not degraded.

Second, dimensionality reduction typically results in models that are easier to understand, which comes in handy when a machine learning expert works in collaboration with a domain expert.

Third, we can use FR as a guidance that reduces our search space which results in much lower costs of the subsequent experiments, for example, when we are trying to search for genetic markers that indicate the presence of a disease.

The last reason was the main motive for the experiments in this paper. Our goal is to establish a small subset of genetic variants that can be used to learn a predictive model that accurately distinguishes between sick and healthy patients. To this end we applied two FR algorithms to the real world problem of discovery of rare genetic variants that play a role in multiple sclerosis (MS) and evaluated their performance.

There is a plethora of FR methods. For their overview, see [8]. The result of applying a FR algorithm to a dataset, is a score $\mathit{impo}(x_i)$, which tells us how much information is contained in the feature x_i with regards to the target y. FR is then obtained by sorting the features with respect to their importance. In this work, we consider the ReliefF [6] and Genie3 [4] FR algorithms.

The rest of the paper is organized as follows. We describe the considered FR algorithms in Section 2. The description of the data and experimental design are presented in Section 3. We present the results in Section 4, and conclude in Section 5.

2. METHODS

In this section, the considered FR algorithms are described. The section starts with the description of the ReliefF algorithm followed by the description of Genie3.

2.1 ReliefF

The motivation behind the ReliefF algorithm is the following. Suppose the instances $\boldsymbol{x}^1$ and $\boldsymbol{x}^2$ are close to each other given some distance measure, but the difference of the corresponding values of a feature x_i is high. If $\boldsymbol{x}^1$ and $\boldsymbol{x}^2$ belong to different classes, we conclude that the change of the values of x_i is one of the reasons for the change of the target value. Hence, x_i has high relevance. However, if $\boldsymbol{x}^1$ and $\boldsymbol{x}^2$ are of the same class, then x_i is not relevant, since the high difference did not cause any change of the target value.

The ReliefF algorithm is an iterative procedure. As can be seen from its pseudocode (Alg. 1), the importances of the features are stored in the list of weights $\boldsymbol{w}$. At each of the m iterations, we randomly select an example $\boldsymbol{x} \in \mathscr{D}_{\text{TRAIN}}$ (line 3) and find its k nearest neighbors of the same class, i.e., *hits* (line 4), and its k nearest neighbours from each of the opposite classes, i.e., *misses* (line 6). The used distance on the descriptive space is the sum of component distances d_i that are defined as

$$d_i(\boldsymbol{x}^1, \boldsymbol{x}^2) = \begin{cases} \mathbf{1}[\boldsymbol{x}_i^1 \neq \boldsymbol{x}_i^2] & : \mathcal{X}_i \text{ nominal} \\ \frac{|\boldsymbol{x}_i^1 - \boldsymbol{x}_i^2|}{\max_{\boldsymbol{x}} \boldsymbol{x}_i - \min_{\boldsymbol{x}} \boldsymbol{x}_i} & : \mathcal{X}_i \text{ numeric} \end{cases}, \qquad (1)$$

At the end of each iteration, the feature importances are updated with the weighted average of the component distances between $\boldsymbol{x}$ and its neighbors.

Algorithm 1 ReliefF($\mathscr{D}_{\text{TRAIN}}$, m, k)

```
1: $\boldsymbol{w} \leftarrow$ zero list of length $D$
2: **for** $j = 1, 2, \ldots, m$ **do**
3:     $\boldsymbol{x} \leftarrow$ random example from $\mathscr{D}_{\text{TRAIN}}$
4:     $H_1, \ldots, H_k \leftarrow k$ nearest hits for $\boldsymbol{x}$
5:     **for all** classes $c \neq \boldsymbol{x}_y$ **do**
6:         $M_{c,1}, \ldots, M_{c,k} \leftarrow k$ nearest misses for $\boldsymbol{x}$ from $c$
7:     **for** $i = 1, 2, \ldots, n$ **do**
8:         $\oplus \leftarrow \sum_{c \neq \boldsymbol{x}_y} \frac{P(c)}{1-P(R_y)} \sum_{l=1}^{k} d_i\left(M_{c,l}, \boldsymbol{x}\right)/mk$
9:         $\ominus \leftarrow \sum_{l=1}^{k} d_i\left(H_l, \boldsymbol{x}\right)/mk$
10:        $\boldsymbol{w}[i] \leftarrow \boldsymbol{w}[i] + \oplus - \ominus$
11: **return** $\boldsymbol{w}$
```

2.2 Genie3

The Genie3 ranking is based on a forest of predictive clustering trees (PCTs) [1, 5] as the baseline classifiers. PCTs generalize decision trees and can be used for a variety of learning tasks, including clustering and different types of prediction. They are induced with the standard top-down induction of decision trees algorithm [2], which takes a set of examples $\mathscr{D}_{\text{TRAIN}}$ as input, and outputs a tree. The heuristic h that is used for selecting the tests in the tree nodes, is the reduction of variance caused by partitioning the instances in a node of the tree. By maximizing the variance reduction, the homogeneity of the instances in the subbranches is maximized: The algorithm is thus guided towards small trees with good predictive performance.

To achieve better predictive performance, one can induce more than one PCT and combine them into an ensemble classifier, called a forest of PCTs. The trees in the forest are not built on a dataset $\mathscr{D}_{\text{TRAIN}}$. Rather, different bootstrap replicates of $\mathscr{D}_{\text{TRAIN}}$ is constructed, for each tree. The prediction of the forest for a given instance $\boldsymbol{x}$ is then typically the class that the majority of the trees voted for.

The main motivation for Genie3 ranking is that splitting the current subset $E \subseteq \mathscr{D}_{\text{TRAIN}}$, according to a test in the node $\mathscr{N}$ where an important feature appears, should result in high variance reduction $h(\mathscr{N})$. Greater emphasis is put on the features higher in the tree where $|E|$ is larger. The Genie3 importance of the feature x_i is defined as

$$impo_{\text{GENIE3}}(x_i) = \frac{1}{|\mathcal{F}|} \sum_{\mathcal{T} \in \mathcal{F}} \sum_{\mathscr{N} \in \mathcal{T}(x_i)} |E(\mathscr{N})| h(\mathscr{N}), \qquad (2)$$

where $\mathcal{T}(x_i)$ is the set of nodes of the tree $\mathcal{T}$ where x_i is part of the test and $E(\mathscr{N})$ is the set of examples that come to the node $\mathscr{N}$.

3. EXPERIMENTAL DESIGN

In this section we present the data used in the experiments that were performed to i) find the locations in human DNA that influence the multiple sclerosis, and ii) check how much different environmental conditions in the data aggregation and processing step influences the results.

3.1 Data Description

Our data consist of 183 instances corresponding to patients. These are divided into three groups: 43 suffering from sporadic multiple sclerosis (SMS), 47 suffering from familial multiple sclerosis (FMS), and 93 being healthy (NoMS). The patients are described by 202487 numeric features which describe the presence of a genetic variant in patients' DNA, and the target variable which describes their diagnosis. The patient genomes were sequenced at the Clinical Institute of Medical Genetics at the University Medical Centre Ljubljana.

Based on the presence of genetic variants on the two strands of DNA, as compared to a reference genome (hg19), we can distinguish between three possible genotypes for every single locus: i) reference sequence on both strands, ii) presence of a genetic variant on one strand only, i.e., in heterozygous state, and iii) presence of a genetic variant on both strands, i.e., in homozygous state. These states are respectively assigned the values 0, 1 and 2. However, the data set contains some missing values, since the success of sequencing and genotyping at a particular locus varies among test subjects.

The feature value for the patients come from two different laboratories: the first and the second gave the results for 171 and 12 patients respectively. Since the different environments could introduce some bias, we prepared two versions of the dataset: one containing all patients and the other, containing only the patients from the first laboratory.

These two versions are used in the experiments where we try to tell apart the three groups of patients (NoMS, SMS and FMS). Following the suggestions of data providers, we also tried to tell apart only healthy and diseased patients. Here, we modify the target variable and merge SPS and FMS into one group (MS). The modified target can now take two different values: MS and NoMS.

3.2 Evaluation Methodology

To assess FR quality, one typically uses k-fold cross-validation (CV), where the data is divided into k parts (folds). At each of k iterations of the procedure, a ranking is constructed from the training set $\mathscr{D}_{\text{TRAIN}}$ which is an union of $k-1$ folds, and then evaluated on the testing set $\mathscr{D}_{\text{TEST}}$, which is the remaining fold. At the end, the per-fold quality measures are aggregated to a single ranking quality score.

This procedure is appropriate if one wants to evaluate the quality of a FR algorithm and is not interested in the actual FR (different FRs correspond to different training folds). This is not the case in this study. On contrary, we are interested in the quality of one particular FR that is to be reported to the domain experts, so we slightly modified the standard evaluation procedure. Taking into account the specifics of the ReliefF and Genie3 FR algorithms, we adopted the following two approaches:

For ReliefF, we use k-fold CV, but we do not evaluate per-fold FRs. Rather, we first average them into one single FR by sorting the features by their average per-fold importances. This average FR is then evaluated in the subsequent steps.

We use an analogous procedure for the Genie3 ranking. Note that the Genie3 importance (Eq. 2) is actually an average of importances for different trees in the forest. Moreover, each tree is built on different bootstrap replicate of the data which does not contain all known examples. This is why we simply run the algorithm on the whole dataset $\mathscr{D}$. The obtained ranking is then evaluated in the subsequent steps.

The remainder of the evaluation procedure is the same for both FR methods and is a variant of the one proposed by Slavkov [7]. Again, we use k-fold CV. At each iteration, we first build a predictive model on a training fold $\mathscr{D}_{\text{TRAIN}}$, considering only the j topmost features of the average ranking, for each value of $1 \leq j \leq D$, such that $j = 1 + \ell(\ell+1)/2$ for some $\ell \in \mathbb{N}$ or $j = D$. Each of these models is then tested on the testing fold $\mathscr{D}_{\text{TEST}}$.

The result of cross-validation are confusion matrices M_j. The (c, d)-th entry of the matrix M_j tells how many patients from the class c were assigned the class d by the classifier that was built from the topmost j features in the ranking.

Let α_j denote the accuracy, computed from the matrix M_j. The points (j, α_j) form a *feature addition curve.* The motivation behind this approach is that for higher α_j's, more relevant features are positioned at the beginning of the FR. Moreover, from the shape of the curve, we can deduce some qualitative characteristics of the FR. E.g., if the curve does not ascend in some part, that means only redundant or irrelevant features are placed in the corresponding part of the FR.

If we want to express the quality of a FR as a single number, we can compute the weighted average α of the accuracies α_j: $\alpha = (\sum_j w_j\alpha_j)/w$, where $w = \sum_j w_j$ and the weights w_j decrease with j, since the beginning of a FR is considered the most important. In our experiments, we choose $w_j = 1/j$, as suggested by Slavkov [7]. A good FR has a high α score.

To asses the influence of different sources of the data, i.e., two laboratories, we use the Jaccard similarity index. For a fixed size j of the set of topmost features, we compute the Jaccard similarity index

$$JSI_j = |B_j \cap F_j|/|B_j \cup F_j| \tag{3}$$

between the sets B_j and F_j that correspond to the rankings computed on a data from both laboratories (B_j) and the first laboratory (F_j). Additionally, we also compute an approximation of the expected value $\widehat{JSI}_j$ of the index between to random feature subsets: $\widehat{JSI}_j = j/(2D - j)$.

3.3 Algorithm Parametrisation

For the ReliefF algorithm, the default values of the parameters were used: the number of iterations was set to $m = |\mathscr{D}_{\text{TRAIN}}|$, and the number of neighbours was set to $k = 10$. To compute the Genie3 ranking, a forest of 1000 trees is grown. The random forest subspace size was set to 25% of the features.

Since the dataset is not too big, leave-one-out CV is used for obtaining the ReliefF ranking, as well as for evaluation of both average rankings. Here, the support vector machines with linear kernel were used as a classifier [3].

4. RESULTS AND DISCUSSION

To asses, which of the FRs found more promising genetic markers in human DNA that influence the MS, we compute the feature addition curves (Sec. 3.2). Fig. 1 shows the results for the first laboratory and binary target, but the graphs for the other three versions of the data are similar.

More specifically, in all four cases the ranking algorithms successfully discover important features, since the curves are ascending in the first part when relevant features are added to feature subsets. Later on, the irrelevant features prevail and the performance slowly decreases (see Fig. 1). Next, at the beginning, Genie3's curve is always clearly above the ReliefF's. Finally, the maximal accuracy of Genie3's ranking is always higher than ReliefF's, and is also achieved sooner.

As a consequence, the α scores of the Genie3 FRs are higher than those of ReliefF, as shown in Tab. 1. This table also shows that the data coming from both laboratories and having binary target, result both in the best FR among all Genie3 FRs, and in the worst FR among all ReliefF FRs.

Table 1: The α scores of the Genie3 and ReliefF rankings, for all versions of the data.

Laboratory	Target values	Genie3	ReliefF
Both	$\{\text{MS}, \text{NoMS}\}$	0.804	0.633
Both	$\{\text{FMS}, \text{SMS}, \text{NoMS}\}$	0.697	0.677
First	$\{\text{MS}, \text{NoMS}\}$	0.753	0.605
First	$\{\text{FMS}, \text{SMS}, \text{NoMS}\}$	0.745	0.711

We inspect the influence of different sources of data by computing the *JSI* (Eq. 3) between the sets of the topmost features of the FRs that base on data from both laboratories and from the first laboratory only. Fig. 2 shows that different sources notably influence the FRs. The fluctuations at the very beginning are expected, since every difference greatly influences the JSI values, when feature subsets are small.

After that, the curve of the ReliefF ranking stabilizes at approximately 0.8 which means that these rankings identify the same features as important. This does not hold in the case of Genie3 rankings. The corresponding sets of 10012

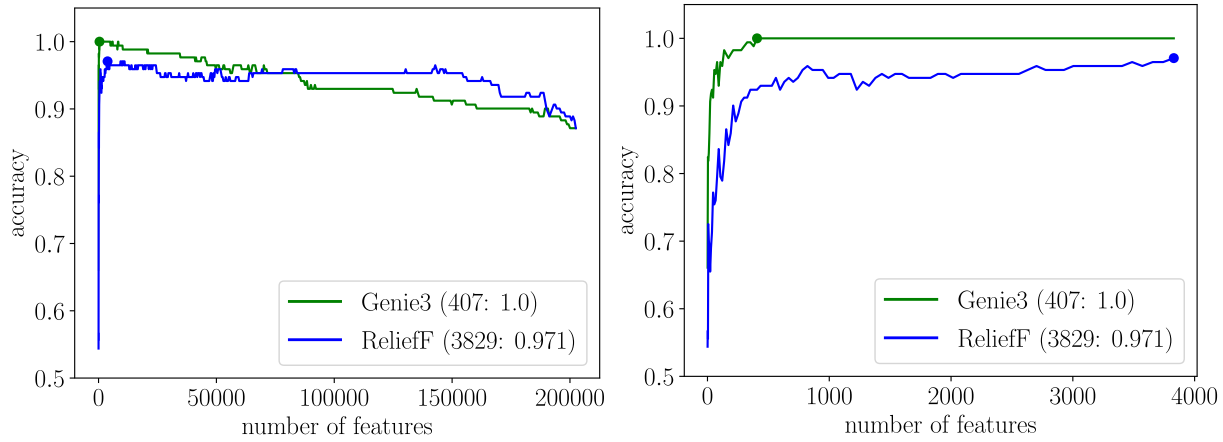


Figure 1: Feature addition curves for the rankings produced by the Genie3 and ReliefF algorithms, using data from the first laboratory only and considering a binary target (left: complete feature addition curves, right: feature addition curves for the first 4000 features). The numbers in the brackets correspond to the maximum accuracy and the number of features where it is first achieved.

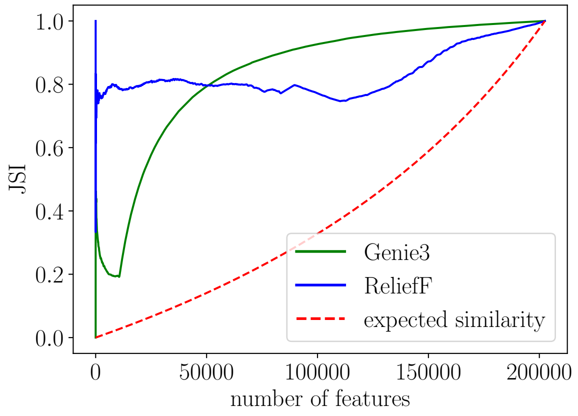


Figure 2: Jaccard similarity of the topmost features of the rankings using the data from the first and both laboratories. The expected similarity corresponds to random ranking and serves as a baseline.

features still have $\widehat{JSI} < 0.2$, which means that different features are recognized as important. Therefore, the Genie3 ranking is more sensitive to changes in the data, since the 12 additional patients from the second laboratory notably changed the rankings. This finding also confirms the data providers' concerns about the influence of different environmental conditions on the data aggregation and processing.

5. CONCLUSIONS

We used the Genie3 and ReliefF algorithm to identify rare genetic variants related to multiple sclerosis. The feature addition curves reveal that the rankings produced by the Genie3 algorithm are better than those of ReliefF, but they are also more sensitive to changes in the data, as shown by low values of the *JSI* score.

However, since the Genie3 algorithm consistently outperformed ReliefF in terms of α scores, only the Genie3 rankings were reported to the domain experts. They further focused and analyzed a small subset of relevant features. They compared the top ranked features to results reported in the literature. In the small subset they found matches to genes that have been reported to be associated with MS. Given the positive matching, additional experimental validation of top ranked features can be performed in order to determine the existence of previously unconsidered causal relations.

We plan to run the algorithms on a new version of the data, processed using a new pipeline that takes into account the different environmental conditions. Since there is a taxonomic relation between the classes, we will also consider hierarchical classification as the baseline for FR.

6. REFERENCES

[1] H. Blockeel. *Top-Down Induction of First Order Logical Decision Trees*. PhD thesis, Katholieke Universiteit Leuven, 1998.

[2] L. Breiman, J. Friedman, R. Olshen, and C. J. Stone. *Classification and Regression Trees*. Chapman & Hall/CRC, 1984.

[3] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46(1):389–422, 2002.

[4] V. A. Huynh-Thu, I. Irrthum, Wehenkel, and P. Geurts. Inferring regulatory networks from expression data using tree-based methods. *PLoS One*, 5(9):1–10, 2010.

[5] D. Kocev, C. Vens, J. Struyf, and S. Džeroski. Tree ensembles for predicting structured outputs. *Pattern Recognition*, 46(3):817–833, 2013.

[6] I. Kononenko and M. Robnik-Šikonja. Theoretical and Empirical Analysis of ReliefF and RReliefF. *Machine Learning Journal*, 55:23–69, 2003.

[7] I. Slavkov. *An Evaluation Method for Feature Rankings*. PhD thesis, International Postgradudate School Jožef Stefan, Ljubljana, 2012.

[8] U. Stańczyk and L. C. Jain, editors. *Feature Selection for Data and Pattern Recognition*. Studies in Computational Intelligence. Springer, 2015.

Modeling of Dynamical Systems: A Survey of Tools and a Case StudY

Gjorgi Peev[1,2] Nikola Simidjievski[1] Sašo Džeroski[1,2]

[1]Jozef Stefan Institute [2]Jozef Stefan International Postgraduate School
Jamova cesta 39, Ljubljana, Slovenia (Emails: Name.Surname@ijs.si)

ABSTRACT

Process-based modeling refers to an approach for automated construction of models of dynamical systems from knowledge and measurements. The underlying formalism allows for both explanatory representation of a dynamical systems in terms of principle system components, as well as their transformation into equations adequate for simulation. The process-based modeling approach, while successful in addressing a variety of modeling tasks, still struggles with meeting some user-interface criteria acceptable for a wider scope of users. In this paper, we review several state-of-the-art approaches and formalisms for (automated) modeling of dynamical systems, and compare them to the most recent implementation of the process-based modeling approach – ProBMoT (Process-based modeling tool).

Keywords

automated modeling, process-based modeling, dynamical systems, formalism, software

1. INTRODUCTION

Models of dynamical systems yield a mathematical representation of the nature laws that govern the behavior of the system at hand. Such models are employed to recreate or simulate the behavior of dynamical systems under diverse conditions.

The two principal elements of every approach to modeling dynamical systems are (1) structure identification and (2) parameter estimation. The former tackles the task of establishing a structure of a model in terms of equations, while the latter deals with approximation of the constant parameters and initial values of the variables in the model for a given structure. Typically, two approaches are being used for modeling dynamical systems: knowledge-driven (white-box) and data-driven (black-box) modeling. The former relates to a domain expert deriving a proper structure of a model by employing extensive knowledge about the system at hand. In turn, the model's parameters are estimated either by using measured data, or manually based on the expert's experience. The latter methodology refers to a trial-error principle: it uses measured data to search for a structure/parameters combination that best fits the observed behavior.

Process-based modeling (PBM) [1,2,3], refers to a grey-box approach, since it joins the knowledge- and data-driven modeling approaches and allows for automated modeling of dynamical systems. In particular, process-based modeling employs both domain-specific knowledge and data for simultaneously constructing the structure of the model and estimating its parameters. The resulting process-based model offers both high-level explanatory representation of a dynamical systems in terms of its principle system components, as well as their transformation into a low-level formalism in terms of equations adequate for simulation of the system's behavior.

The latest implementation of the process-based modeling paradigm – ProBMoT [4,5] uses text-based, non-visual formalism, which presents a challenge when it comes to visualizing the structure of the modeled system. While scientists can typically comprehend and relate to models formalized as equations, the (uncommon) high-level PBM formalism is not always familiar to them. Currently, this makes ProBMoT usable for a narrow scope of domain experts. On the other hand, several state-of-the art grey-box modeling software such as: Prometheus [6], Eureqa [7], MATLAB [8], STELLA [9] and COPASI [10] have been used extensively for different modeling tasks in a variety of domains.

In order to widen ProBMoT's user base, in this paper we aim at identifying the main features and limitations of each of the aforementioned modeling software and compare them to ProBMoT. In particular, we attempt at modeling a two-cascaded water tanks system, a well-known system identification benchmark, with each of the six modeling tools and compare them in terms of their input and output according to several criteria. However, quantifying and describing a modeling software and its formalism, is not a trivial task. To this end, we propose five criteria according to which we survey the different modeling approaches:

(**C1**) *Generality:* the applicability of a software to a general problem (from all fields). In contrast, there are software applicable to problems from specific fields (molecular biology, finances, ecology, electronics, etc.).

(**C2**) *Parameter estimation:* capability of fitting the model's parameter values to data.

(**C3**) *Automated modeling*: ability to learn models with automated computational scientific discovery methods. Note that, here we can distinguish also between fully automatic and semi-automatic approaches. The former does not rely on prior knowledge about a domain and typically results in one model structure built from scratch. The latter refers to the ability to discover a set of explanatory models blending expert domain knowledge with computational discovery algorithms.

(**C4**) *Graphical representation*: ability to graphically represent the output models of the software.

(**C5**) *Comprehensibility:* whether the output of the software is comprehensible on first hand to the domain-expert user, without the need of background knowledge.

The rest of this paper is organized as follows. In the next section we outline the six state-of-the-art tools for modeling dynamical systems. Section 3 elaborates the design of the modeling experiment and presents the results, which in turn are discussed in Section 4. Finally, Section 5 concludes the paper.

2. BACKGROUND

In this section, we focus on six grey-box modeling software packages and their characteristics: ProBMoT, Prometheus, Eureqa, MATLAB, STELLA and COPASI.

ProBMoT [4,5] (Process-Based Modeling Tool) is the latest implementation of the process-based modeling paradigm. It is a software for construction, parameter estimation and simulation of process-based models.

The output of this software is a process-based model, represented with entities and processes. Entities relate to the actors of the observed system, defined with constants and variables. The processes, on the other hand, represent the interactions between the entities, referring to one or more ordinary/algebraic equation. Collating the equations from all the processes in the model, a system of ODEs can be attained.

To this end, ProBMoT takes as input a library of domain knowledge, a task specification and data.

The library is formalized by establishing templates of generic entities that appear in the generic processes. The templates can be organized into a hierarchical structure. The task specification limits the search space of candidate model structures by supplying constraints, specified as incomplete conceptual models as modeling presumptions. The library of domain-specific knowledge together with the task specification determine the space of models. The induction algorithm then searches through this space of candidate model structures, finding plausible model solutions and estimating the constant parameters of each candidate model structure to the input data.

Prometheus [6] is a software that supports interaction between the user and computational discovery algorithms. The formalism used in Prometheus specifies process models and background knowledge in terms of variables and processes that relate them. Each process express casual relations between its input and output variables through one or more differential equations.

The input for this software is a user-defined model, library of background knowledge consisted of generic processes, measured data and constraints specifying what can be revised. The output is a revised model-structure that best fits the measured data.

Prometheus is a predecessor to ProBMoT, and consequently their formalisms are comparable. Prometheus uses process models, which are analogous to ProBMoT's process-based models. Their definitions for processes as model's components are similar. The difference is that the variables in Prometheus are not encoded in an entity, but they are represented as a component.

Eureqa [7] uses symbolic regression [11] with genetic programming in order to infer equation-based structure of the system and its parameters solely from data, by minimizing the error using the implicit derivatives method. The state of the modeled system is declared with a target variable, its descriptors and their form in order to define the search space. Note that, the modelers have little-to-no control over the space of plausible structures. This means that, it is still a domain expert's task to infer the similarities between the resulting model and the real system structure.

MATLAB [8] supplies functions for performing system identification and parameter tuning of a user pre-defined model. Its formalism allows specifying quantitative models with instantaneous and differential equations. Note that, MATLAB does not support automated modeling, i.e., learning multiple structures.

The models are defined as model objects, i.e. specialized data containers that encapsulate data and other model attributes. The dynamics of the system at hand are described with ODEs imported in a C MEX-file. With associating the model object to the C MEX-file, and employing functions for simulation and parameter estimation, on the output MATLAB obtains a completely defined model structure with all parameters tuned in accordance to data.

STELLA's [9] formalism relates to stocks, flows, convertors and connectors. Stocks represent variables, flows denote their changes over time (derivatives), converters encode the constant parameters, while connectors are used to attain a link between all of them. While the models are built using this formalism, the software produces finite difference equations that describe it. Note that, this formalism is comparable to the PBM. We can associate stocks with entities, and flows with processes. Similar to MATLAB, the input to STELLA is a user-specified quantitative model and measured data. Similarly, STELLA also does not support automated modeling, therefore the output model structure is never learned. However, one can still simulate the complete model by invoking the simulator that can run the input user-defined model.

COPASI [10] is a software for simulation and analysis the dynamics of biochemical networks. It supports models in the SBML standard [12]. The models are defined with chemical reactions between molecular species. They can also include compartments, events, and other global variables that can help specify the dynamics of the system. Here, we can also draw an analogy between COPASI's and ProBMoT's formalisms. The species in COPASI correspond to ProBMoT's entities, while reactions are analogous to processes containing equations which describe the behavior of the system. However, in contrast to ProBMoT, COPASI does not perform automatic structure identification. The input to COPASI is user-defined model and measured data. The result is a complete model, with parameters tuned to best fit the data.

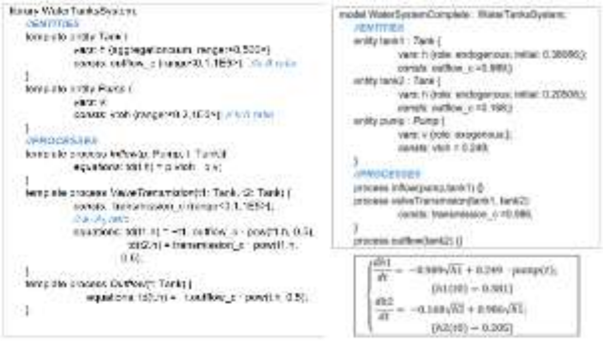
3. CASE STUDY

In order to better illustrate and evaluate the formalisms of the software described in the previous section, here we tackle the task of modeling a two-cascaded water tanks system [13]. The system is consisted of two cascaded water tanks with free outlets, fed by a pump. The governing equations for this system are depicted below (Eq. 1), where the states of the water levels of the two tanks are denoted with h_1 and h_2, the latter (h_2) being the output. The voltage applied to the pump is u(t), while A_1, A_2, a_1 and a_2 denote the areas of the tanks and their effluent areas, while the applied voltage-to-flow conversion constant is denoted with *k*. The task is to model the response of the lower tank.

$$\begin{cases} \frac{dh_1}{dt} = -\frac{a_1\sqrt{2g}}{A_1}\sqrt{h_1} + \frac{k}{A_1}u(t) \\ \frac{dh_2}{dt} = -\frac{a_2\sqrt{2g}}{A_2}\sqrt{h_2} + \frac{a_1\sqrt{2g}}{A_2}\sqrt{h_1} \end{cases}$$

Equation 1. Two-cascaded water tanks system

ProBMoT - In order to model the water tanks system in ProBMoT, we first need to create a library of domain knowledge (Figure 1A), i.e. we need to formulate template entities and processes which in turn will be instantiated to specific entities and processes. The main actors in the system are two water tanks (with same properties) and a pump. In terms of template entities, this translates to one template entity *Thank* and a template entity *Pump.* The tanks are characterized with a variable *h*, representing the water level height, and a constant *outflow_c* which denotes the ration between the tanks areas (a/A). The dynamics that govern the system's behavior in terms of equations are encoded in template processes *inflow*, *ValveTransmission, outflow*. These correspond to the water inflow in the first tanks, the water flows between the two tanks, and the water outflow from the second tank, respectively. In terms of defining modeling constraints, we can outline the number of entities involved in the system and encode the plausible process alternatives. In turn, such a library together with the modeling constraints, can be induced to a specific model structure (Figure 1B-top) of the particular system, parameters of which are fitted using the measured data. Such a model can be then transformed into to a system of ODEs (Figure 1B-bottom) and simulated.



```
library WaterTanksSystem;
  //ENTITIES
  template entity Tank {
        vars: h {aggregation:sum, range:<0,500>};
        consts: outflow_c {range:<0,1.1E5>}; [illegible]
  }
  template entity Pump {
        vars: v;
        consts: vtoh {range:<0.2,1E5>}; [illegible]
  }
  //PROCESSES
  template process Inflow(p: Pump, t: Tank){
        equations: td(t.h) = p.vtoh * p.v;
  }
  template process ValveTransmission(t1: Tank, t2: Tank) {
        consts: transmission_c {range:<0,1.1E5>};
        [illegible]
        equations: td(t1.h) = -t1.outflow_c * pow(t1.h, 0.5),
                   td(t2.h) = transmission_c * pow(t1.h,
                  0.5);
  }
  template process Outflow(t: Tank) {
        equations: td(t.h) = -t.outflow_c * pow(t.h, 0.5);
  }
```

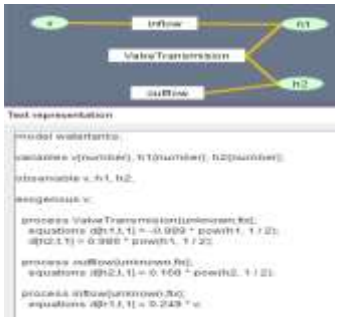
```
model WaterSystemComplete : WaterTanksSystem;
  //ENTITIES
  entity tank1 : Tank {
        vars: h {role: endogenous; initial: 0.38086};
        consts: outflow_c = 0.989;}
  entity tank2 : Tank {
        vars: h {role: endogenous; initial: 0.20508};
        consts: outflow_c = 0.168;}
  entity pump : Pump {
        vars: v {role: exogenous};
        consts: vtoh = 0.249;
  }
  //PROCESSES
  process Inflow(pump,tank1) {}
  process valveTransmission(tank1, tank2){
        consts: transmission_c = 0.986;
  }
  process outflow(tank2) {}
```

$$\begin{cases} \frac{dh1}{dt} = -0.989\sqrt{h1} + 0.249 \cdot pump(t); \\ \quad (h1(t0) = 0.381) \\ \frac{dh2}{dt} = -0.168\sqrt{h2} + 0.986\sqrt{h1}; \\ \quad (h2(t0) = 0.205) \end{cases}$$

A) *B)*

Figure 1. A) Library of background knowledge for a water tank system B) Process-based model of a particular water-tank system (top); The same model transformed to ODEs (bottom)

Prometheus - In a similar fashion to ProBMoT, we define processes with equations in Prometheus as well. We have process *inflow*, *valvetransmition* and *outflow*. In this formalism, entity components are not present, but the variables represent a component themselves. Consequently, we have three variable components: *h1*, *h2* (as observable), and *v* (as exogenous). We create a library of background knowledge, containing generic processes, where the equations instead of numbers have parameters, and with the measured data, we refine the model structure. We obtain the final defined model structure with estimated parameters. The models obtained in Prometheus (Figure 2) are highly comprehensible, since the software has visual representation for its models.



```
model watertanks;

variables v{number}, h1{number}, h2{number};

observable v, h1, h2;

exogenous v;

  process ValveTransmission(unknown.fx);
    equations d[h1,t,1] = -0.989 * pow(h1, 1 / 2);
    d[h2,t,1] = 0.986 * pow(h1, 1 / 2);

  process outflow(unknown.fx);
    equations d[h2,t,1] = 0.168 * pow(h2, 1 / 2);

  process inflow(unknown.fx);
    equations d[h1,t,1] = 0.249 * v;
```

Figure 2. Prometheus model of a particular water tank system

Eureqa - For the two-tank system, we define $\frac{dh2}{dt}$ as the target variable. The form of the equation that describes our target should use *h1*, *h2*, $\frac{dh1}{dt}$ *and v* as descriptive variables. During the search, we can visualize (Figure 3) the found equations, ranked on a complexity-error graph. If the error is not decreasing significantly, and the complexity is increasing, we can stop the search at any time. In the end, it results with a set of equations, from which we choose the most suitable one, obtaining a complete model for our water tanks system.

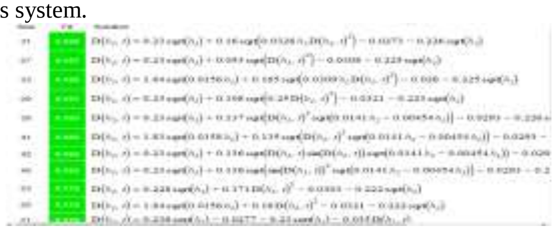
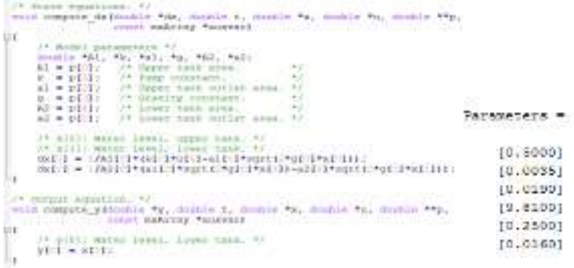


Figure 3. Eureqa model of a particular water tank system

MATLAB - We describe the dynamics of the two-tank system with writing the two differential equations that govern its behavior into a C MEX-file. First, we initialize all the parameters that we are going to use for modeling. Second, we write a function (*compute_dx)*, which computes the state equations, i.e. the change in the water height level of the two tanks over time ($\frac{dh1}{dt}$ and $\frac{dh2}{dt}$, represented as dx[0] and dx[1] respectively). Next, we write a function (*compute_y)*, which computes the output equation, in our case the response of the lower tank, i.e. y[0] = x[1]. In turn, with the *Identify non-linear grey box model* function in MATLAB, we associate a model object with the C MEX-file, resulting in a grey-box model of the system at hand (Figure 4). We choose which of the parameters described we want to estimate, and with the function *Non-linear grey-box estimate* they are fitted to the data provided at input. The obtained model can then be simulated.

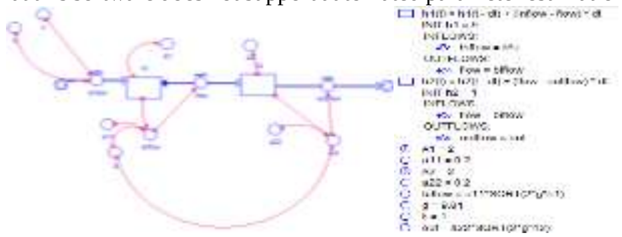


```
/* State equations. */
void compute_dx(double *dx, double t, double *x, double *u, double **p,
                const mxArray *auxvar)
{
    /* Model parameters */
    double *A1, *k, *a1, *g, *A2, *a2;
    A1 = p[0];   /* Upper tank area.            */
    k  = p[1];   /* Pump constant.              */
    a1 = p[2];   /* Upper tank outlet area.     */
    g  = p[3];   /* Gravity constant.           */
    A2 = p[4];   /* Lower tank area.            */
    a2 = p[5];   /* Lower tank outlet area.     */

    /* x[0]: Water level, upper tank. */
    /* x[1]: Water level, lower tank. */
    dx[0] = 1/A1[0]*(k[0]*u[0]-a1[0]*sqrt(2*g[0]*x[0]));
    dx[1] = 1/A2[0]*(a1[0]*sqrt(2*g[0]*x[0])-a2[0]*sqrt(2*g[0]*x[1]));
}

/* Output equation. */
void compute_y(double *y, double t, double *x, double *u, double **p,
               const mxArray *auxvar)
{
    /* y[0]: Water level, lower tank. */
    y[0] = x[1];
}
```

```
Parameters =

[0.5000]
[0.0035]
[0.0190]
[9.8100]
[0.2500]
[0.0160]
```

Figure 4. Structure of a C MEX-file with and output equation and estimated parameters

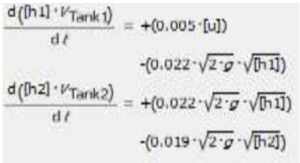
STELLA - We model the particular system at hand with two stocks as the main actors of the system, representing the two water tanks. The change in their water height is indicated with three flows. *Flow #1* represents the amount of water transferred from the upper to the lower tank: the outflow of the upper is an inflow for the lower tank. *Flow #2* is the outflow from the lower tank and *Flow #3* is inflow from the pump into the upper tank. The other components and parameters are represented with convertors. The model obtained in STELLA is graphically highly comprehensible. The drawback is that this software does not support automated parameter estimation.



```
h1(t) = h1(t - dt) + (inflow - flow) * dt
  INIT h1 = [illegible]
  INFLOWS:
    inflow = [illegible]
  OUTFLOWS:
    flow = inflow
h2(t) = h2(t - dt) + (flow - outflow) * dt
  INIT h2 = 1
  INFLOWS:
    flow = inflow
  OUTFLOWS:
    outflow = [illegible]
A1 = 2
a11 = 0.2
A2 = 2
a22 = 0.2
inflow = [illegible]
g = 9.81
k = 1
out = [illegible]
```

Figure 5. STELLA model of a particular water tank system

COPASI - We compose our model with three compartments, species and reactions (Figure 6). The lower, the upper tank and the environment (in which the two tanks are in) represent different compartments themselves. Compartment *Tank1* has initial expression of the area of the upper tank (*A1*). Similarly, *Tank2* has initial expression of the area of the lower tank (*A2*). In similar fashion to ProBMoT, where each entity has variables or constants, here each compartment contains species: *Tank1* includes height *h1*, *Tank2* holds height *h2* and *Environment* contains the voltage *u* applied to the pump. Reactions in COPASI are analogous to ProBMoT's processes, and knowing the equations from (Eq.1), we define reaction *flow* (between *h1* and *h2*) reaction *inflow* (between *u* and *h1*) and reaction *outflow* (from *h2*). As an output, we obtain a model of differential equations with computed parameter values. COPASI is widely used in the field of biochemical networks and their dynamics. However, using COPASI outside of that fields, as the case in this paper, is not a trivial task.



$$\frac{d([h1]\cdot V_{Tank1})}{dt} = +(0.005\cdot[u]) -(0.022\cdot\sqrt{2\cdot g}\cdot\sqrt{[h1]})$$

$$\frac{d([h2]\cdot V_{Tank2})}{dt} = +(0.022\cdot\sqrt{2\cdot g}\cdot\sqrt{[h1]}) -(0.019\cdot\sqrt{2\cdot g}\cdot\sqrt{[h2]})$$

Figure 6. COPASI model of a particular water tank system

4. DISCUSSION

Having the identified the key characteristics of each of the six modeling software, here we compare them based on the criteria defined in Section 1. Table 1 presents the results of the study.

Table 1. Comparing the different modeling tools according the five different cirteria

	C1	C2	C3	C4	C5
ProBMoT	✓	✓	✓	✗	✗
Prometheus	✓	✓	✓	✓	✓
Eureqa	✓	✓	✓	✗	✓
MATLAB	✓	✓	✗	✗	✗
STELLA	✓	✗	✗	✓	✓
COPASI	✗	✓	✗	✗	✓

Based on the first criterion ***(C1 - generality)*** all software except COPASI are general-purpose, meaning that systems from different fields can be easily modeled and simulated with them. COPASI is specific-purpose software, built around the logic of the biochemical networks and their dynamics. This means building every type of model in COPASI can be very challenging and ambitious.

According to the second criterion ***(C2 - parameter estimation)*,** all tools except STELLA have integrated parameter estimation methods. In STELLA, the parameters are tuned manually.

In terms of *automated modeling* ***(C3)*,** ProBMoT, Prometheus and Eureqa are capable of automated modeling. Eureqa is able to infer equation-based model from scratch using genetic programing. On the other hand, ProBMoT and Prometheus are able to find set of models with similar structure components and distinguish among them. Both of them relay on domain-specific modeling knowledge used in the process of induction of model structures.

In terms of *graphical representation* ***(C4)***, only STELLA and Prometheus have the ability to graphically visualize the models and their components, in the form of building blocks.

Finally, in terms of *comprehensibility* ***(C5)*,** the high-level modeling formalisms used by ProBMoT and Prometheus results in not widely interpretable models. Still, both have the ability to transform the high-level formalism in equations, with additionally Prometheus having the ability to visualize the modeling components. While MATLAB is widely used for modeling tasks, its formalism still requires a low-level programing knowledge for one to be able to encode and decode the models. Regarding Eureqa, while the output models are contained of (differential) equations, the model structure doesn't necessarily correspond to the real system's structure. The obtained models in COPASI are comprehensible for experts in the field of biochemical networks. Finally, the models obtained from STELLA besides being offering graphical visualization the models can also be translated into equations. Either way, it is highly interpretable.

5. CONCLUSION

In this paper, we give an overview of a formalism for automated modeling of dynamical systems, named process-based modeling. It is not always easily interpretable, making it usable only for a narrow scope of domain experts. In order to improve that, we review several state-of-the art grey-box modeling tools, identifying their main features and limitations. As a case study, we model a two-water-tanks system with all the different tools and compare them in terms of their input and output according to five criteria.

The general conclusion of this paper is that ProBMoT, the latest software implementation of the PBM paradigm, while successful in tackling various modeling tasks, is usable for a very narrow scope of users mainly because of its uncommon high-level modeling language and the lack of graphical representation of the resulting models. We conjecture that establishing a Graphical User Interface (GUI) for it, will address its usability issues.

However, in order to create a GUI for ProBMoT, we first need to address the non-trivial and abstract problems of visually representing the hierarchical nature of the process-based models.

One answer could be presenting the components of the process-based models as building blocks, similar to the model components employed in STELLA or Prometheus. It would enable the user to make libraries and define tasks graphically and interactively.

Another feature that could be of good use for the GUI is tightly connected with the runtime process, the results and data visualization. Similarly to Eureqa, during the search, all found feasible models can be listed and ranked.

To conclude, with developing a self-explanatory visual representation of the process-based modeling formalism, comprehensible for domain-expert scientists, the PBM paradigm would become more approachable. With a universal visual representation, scientists from different fields would be able to transfer knowledge between them.

Acknowledgements. We would like to acknowledge the support of prof. dr. Ljupco Todorovski.

6. REFERENCES

[1] Džeroski, S., Todorovski, L. (2003). Learning population dynamics models from data and domain knowledge. ***Ecological Modelling***, 170(2-3):129-140, Elsevier.

[2] Langley, P., Saanchez, J., Todorovski, L., Džeroski, S. (2002). Inducing process models from continuous data. *Proc. 19th International Conference on Machine Learning 2002*, pp. 347- 354

[3] Todorovski, L., Džeroski, S. (2006). Integrating knowledge-driven and data-driven approaches to modeling. ***Ecological Modelling***, 194(1-3):3-13, Elsevier.

[4] Čerepnalkoski, D. (2013). Process-based models of dynamical systems: Representation and induction, PhD Thesis, Jožef Stefan International Postgraduate School, Ljubljana, Slovenia.

[5] Tanevski, J., Simidjievski, N., Todorovski, L., Dzerovski, S. (2017). Process-based Modeling and Design of Dynamical Systems. *Proc. Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2017.*

[6] Sanchez, J.N., Langley, P. (2003). An interactive environment for scientific model construction, *Proc. 2nd international conference on Knowledge capture, K-CAP'03*, pp. 138-145, ACM.

[7] Schmidt, M., Lipson, H. (2009). Distilling Free-Form Natural Laws from Experimental Data. ***Science*** 324(5923):81-85, AAAS.

[8] MathWorks, Inc. (2004). *MATLAB: the language of technical computing: computation, visualization, programming.* USA.

[9] Richmond, B. (1985). STELLA: Software for Bringing System Dynamics to the Other 98%. *Proc. 3rd International Conference of the System Dynamics Society*, pp. 706–718

[10] Hoops, S., Sahle, S., et. al. (2006). COPASI: a COmplex PAthway SImulator. ***Bioinformatics*** 22(24):3067-3074

[11] Koza, J.R. (1992). *Genetic Programming: On the programming of computers by means of natural selection*. MIT Press, USA

[12] Hucka, M., Finney, A., et. al. (2003). The systems biology markup language (SBML): A medium for representation and exchange of biochemical network models. ***Bioinformatics*** 19 (4):524–531, Oxford University Press.

[13] Wigren, T., Schoukens, J. (2013). Three free data sets for development and benchmark in nonlinear system identification. *Proc. European Control Conference 2013*, pp. 2933–2938, IEEE.

Vpliv različnega prenosnega kanala pri referenčnih in testnih posnetkih na forenzično verifikacijo govorcev

Tomaž Šef
Institut "Jožef Stefan"
Jamova cesta 39
1000 Ljubljana
+386 1 477 34 19
tomaz.sef@ijs.si

Robert Blatnik
Institut "Jožef Stefan"
Jamova cesta 39
1000 Ljubljana
+386 1 477 32 69
robert.blatnik@ijs.si

POVZETEK

V članku obravnavamo problematiko razpoznavanja oz. verifikacije govorcev v forenzične namene in vpliv različnih načinov zajemanja govornega signala na rezultate izvedenih analiz.

Izvedli smo poizkuse s pomočjo komercialnega sistema za samodejno razpoznavanje govorcev (SRG) in preučevali razlike v njegovi uspešnosti glede na različne kombinacije prenosnih kanalov pri zajemanju učnih oz. referenčnih in testnih posnetkov.

Podani so rezultati eksperimentov za slovenske govorce, ki smo jih simultano snemali preko petih različnih prenosnih kanalov in ob treh različnih načinih govorjenja: branje, spontani govor in dialog.

Ključne besede

Forenzične analize, razpoznavanje oz. verifikacija govorcev, prenosni kanali za zajemanje govora, govorna zbirka.

1. UVOD

Pri razpoznavanju oz. verifikaciji govorcev v forenzične namene imamo opravka s spornimi posnetki izgovarjav, ki predstavljajo dokazno gradivo in so posneti v »stvarnih pogojih« med samim izvajanjem kaznivih dejanj. V večini primerov govorni posnetki predstavljajo telefonske pogovore, pridobljene predvsem na dva načina: (i) anonimen klic, kadar je pričakovan ali kako drugače dostopen, (ii) prisluškovanje telefonskim pogovorom s strani policije. Pojem »stvarni pogoji« uporabljamo kot nasprotje »laboratorijskim pogojem«, ko ne moremo nadzirati, pričakovati ali predvidevati pogojev v katerih se bodo pridobili posamezni govorni posnetki. Celo več; obtoženec ponavadi ne želi korektno sodelovati in skuša ovirati ali preprečiti pridobitev kakršnihkoli zanj obremenilnih informacij.

Zaradi »stvarnih pogojev« pridobivanja posnetkov je govorni signal bolj spremenljiv oz. variabilen. Vire variabilnosti govornega signala lahko razvrstimo v naslednje kategorije [1]:

(i) svojske variabilnosti govornih signalov istega govorca: vrsta govora, staranje, časovni presledek med dvema posnetkoma, narečje, žargon, socialni status, čustveno stanje, uporaba omamnih sredstev itd.

(ii) izsiljene oz. umetne variabilnosti govornih signalov istega govorca: »Lombardov« učinek, stres zaradi zunanjega vpliva, »cocktail-party« učinek itd.

(iii) zunanja variabilnost odvisna od kanala: tip telefona ali mikrofona, fiksna/mobilna telefonija, komunikacijski kanal, pasovna širina, dinamični obseg oz. razpon, električni in akustični šum, odmev, popačenje itd.

Forenzični pogoji so doseženi, ko se dejavniki variabilnosti, ki predstavljajo t.i. »stvarne pogoje«, pojavljajo brez kakršnegakoli principa, pravila ali norme. Lahko so konstantni preko celotnega klica ali pa se hipoma pojavijo ali izginejo; na celoten proces vplivajo povsem nepredvidljivo.

2. METODE IDENTIFIKACIJE OZ. VERIFIKACIJE GOVORCEV

Različne metode identifikacije govorcev so lahko bolj ali manj subjektivne oz. objektivne. Tudi pri objektivnih metodah imamo opraviti z določenim vplivom človeka; npr. računalnik je sprogramiran, rezultati pa so interpretirani s strani eksperta. Najbolj subjektivna metoda identifikacije govorcev v forenzične namene je slušno-zaznavna metoda oz. slušna analiza. Nekoliko bolj objektivna je slušno-instrumentalna metoda. Med najbolj objektivne štejemo polavtomatske in avtomatske metode identifikacije govorcev.

Slušno-zaznavna metoda (angl. »aural-perceptual approach«) oz. slušna analiza (angl. »auditory analysis«) v osnovi temelji na pozornem poslušanju posnetkov s strani izkušenega fonetika, pri čemer se zaznane razlike v govoru uporabijo za ocenjevanje stopnje podobnosti med glasovi. Slušna analiza ima svoje omejitve in se pri običajni fonetični analizi uporablja predvsem za izluščenje zanimivih lastnosti in parametrov, ki jih nato podrobneje analiziramo s slušno-instrumentalno metodo [2, 3].

Slušno-instrumentalna metoda (angl. »auditory instrumental approach) vključuje meritve različnih parametrov, kot so npr. osnovna frekvenca (F0), hitrost govora, potek osnovnega tona, razne spektralne karakteristike govornega signala itd. Parametri se nato medsebojno primerjajo po srednjih ali povprečnih vrednostih in variancah. Pri računalniški akustični analizi (angl. »computerised acoustic analysis) dobimo numerične vrednosti različnih govornih parametrov s pomočjo posebne programske opreme. Pri tem je vloga eksperta še vedno zelo pomembna, saj se je potrebno odločiti, kateri govorni vzorci so dovolj dobre kvalitete za analizo. Poleg tega je potrebno izbrati oz. določiti primerljive dele govornih vzorcev, ki bodo analizirani, in ovrednotiti dobljene rezultate. Parametri pri akustično forenzični analizi večinoma izvirajo iz lingvistično-fonetičnih raziskav in so neposredno povezani s slišnimi fonetičnimi značilnostmi [4].

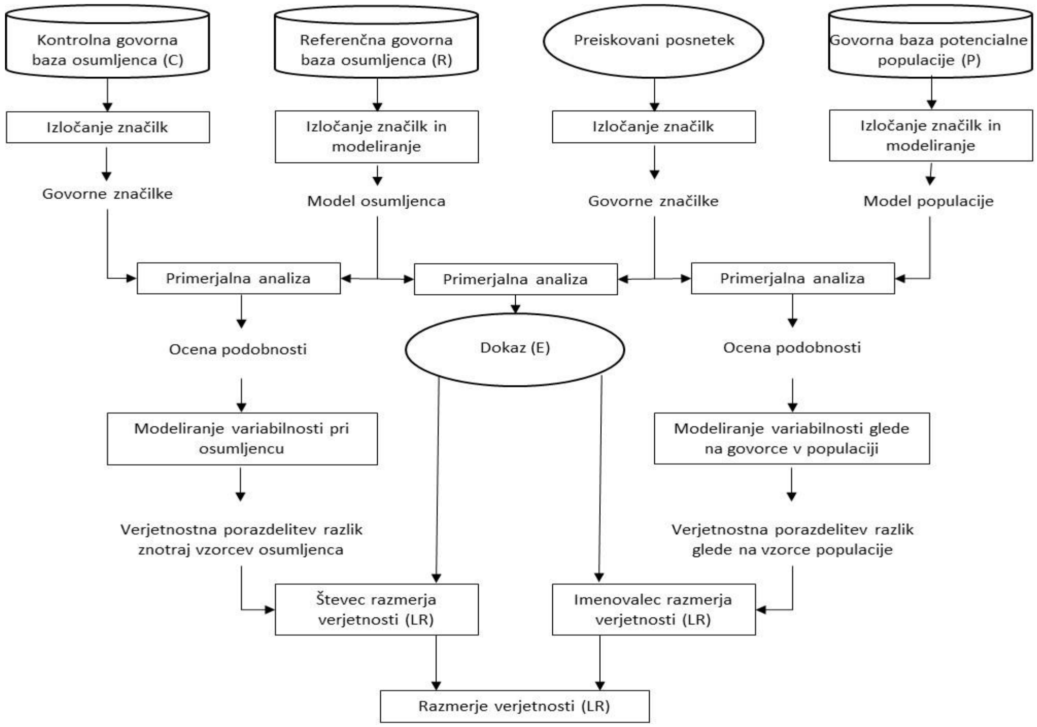
Polavtomatsko (angl. »forensic semiautomatic speaker recognition«) in avtomatsko (angl. »forensic automatic speaker recognition«) razpoznavanje govorcev v forenzične namene je uveljavljen termin za metode (pol)avtomatskega razpoznavanja govorcev, ki so prilagojene za uporabo v forenzične namene. Pri polavtomatskih metodah prihaja med preiskavo do interakcije eksperta in računalnika. Pri avtomatskem razpoznavanju govorcev pa se medsebojno primerjajo statistični modeli akustičnih parametrov glasov znanih govorcev (iz govorne baze) s statističnim modelom akustičnih parametrov nepoznane osebe, ki jo želimo identificirati (slika 1). Na podlagi te primerjave izračunamo kvantitativno oceno podobnosti med (od govorca odvisnimi) parametri glasu nepoznane osebe na posnetku in parametri obdolženca s čimer ocenimo prepričljivost dokaza. Pri avtomatskem razpoznavanju govorcev (slika 2) v forenzične namene je prepričljivost dokaza odvisna od relativne verjetnosti, da opazimo neke značilnosti nepoznanega glasu v statističnem modelu akustičnih parametrov obdolženca in v statističnih modelih glasov potencialne populacije.

Podatkovno vodena Bayesova metoda za avtomatsko razpoznavanja govorcev zahteva poleg preiskovanega posnetka (oz. sledi) uporabo še treh baz izgovarjav (slika 1): referenčno govorno bazo osumljenca (R), ki služi izdelavi statističnega modela njegovega glasu (pogoji snemanja morajo biti čim bolj podobni pogojem pri snemanju govorne baze populacije P), kontrolno govorno bazo osumljenca (C), ki služi ocenjevanju notranje variabilnosti glasu osumljenca (pogoji snemanja morajo biti čim bolj podobni pogojem snemanja preiskovanega posnetka) in govorno bazo potencialne populacije (P), ki vsebuje takšne posnetke glasov, da nobeden naključno izbran posnetek iz te baze ni izgovorjen s strani iste osebe, kot je preiskovani posnetek (oz. sled). Kakršno koli neujemanje govornih baz zaradi okoliščin pri prenosu govornega signala, vrste snemalne naprave, šuma, jezikoslovne vsebine in trajanja posnetkov lahko vpliva na zanesljivost dobljenih rezultatov [5].

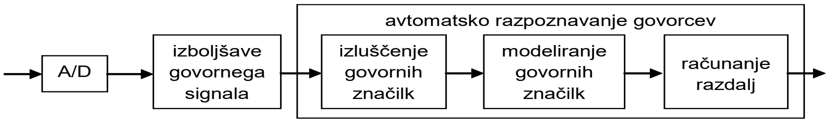
Zadnje čase temeljijo sistemi za (pol)avtomatsko razpoznavanje govorcev v forenzične namene na oceni kvocienta verjetnosti (angl. »likelihood ratio«; LR) [6]. Razmerje verjetnosti (LR) je podano kot razmerje gostote verjetnosti porazdelitev razlik znotraj vzorcev osumljenca in porazdelitev razlik glede na vzorce populacije v točki E (dobimo jo s primerjavo preiskovanega posnetka in statističnega modela osumljenca) [5].

Metode razpoznavanja govorcev, ki temeljijo na tehnikah statističnega modeliranja, kot npr. Gaussov mešani model (angl. Gaussian Mixture Modell, GMM), imajo to dobro lastnost, da neposredno vrnejo verjetnost, ali posamezna izgovorjava lahko pripada statističnemu modelu govorca. Namesto GMM lahko za razpoznavanje govorcev uporabimo tudi prikrite Markovove modele ali nevronske mreže [5]. So pa te metode manj uporabne v forenzičnih postopkih, ker nam določene verjetnosti v praksi praviloma niso poznane in posledično ne moremo izračunati razmerja verjetnosti (LR), ki je najpogosteje uporabljan oz. edini sprejemljiv način podajanja rezultatov na sodiščih [6].

Avtomatski sistemi za razpoznavanje govorcev se ne smejo uporabljati samostojno pač pa le kot dopolnitev drugih metod; sicer obstaja možnost napačne identifikacije [7]. Rezultate različnih forenzičnih metod v praksi preučujemo povezano s čimer dobimo kombinirano oceno zanesljivosti dokaznega gradiva.


Kontrolna govorna baza osumljenca (C)
Referenčna govorna baza osumljenca (R)
Preiskovani posnetek
Govorna baza potencialne populacije (P)
Izločanje značilk
Izločanje značilk in modeliranje
Izločanje značilk
Izločanje značilk in modeliranje
Govorne značilke
Model osumljenca
Govorne značilke
Model populacije
Primerjalna analiza
Primerjalna analiza
Primerjalna analiza
Ocena podobnosti
Dokaz (E)
Ocena podobnosti
Modeliranje variabilnosti pri osumljencu
Modeliranje variabilnosti glede na govorce v populaciji
Verjetnostna porazdelitev razlik znotraj vzorcev osumljenca
Verjetnostna porazdelitev razlik glede na vzorce populacije
Števec razmerja verjetnosti (LR)
Imenovalec razmerja verjetnosti (LR)
Razmerje verjetnosti (LR)

Slika 1. Shematski prikaz izračuna razmerja verjetnosti (LR) pri razpoznavanju govorcev.


avtomatsko razpoznavanje govorcev
A/D
izboljšave govornega signala
izluščenje govornih značilk
modeliranje govornih značilk
računanje razdalj

Slika 2. Postopek avtomatskega razpoznavanja govorcev.

3. GOVORNA ZBIRKA

Glede na omejeno količino urejenih in tehnično primernih govornih posnetkov v slovenskem jeziku smo se odločili za izvedbo snemanja lastne govorne zbirke slovenskih govorcev.

Govorno zbirko smo posneli v laboratoriju. Posneli smo 25 moških slovensko govorečih oseb. Izbrali smo govorce različnih starosti, vse delovno aktivne, v starostni skupini od približno 25 do 65 let.

Večji del posnetkov so govorci hkrati govorili v dva namizna mikrofona (v oddaljenosti 15 do 30 cm od ust govorca), v prostoročni mikrofon VoIP telefona ter v GSM telefon in slušalko klasičnega analognega PSTN telefona. Na ta način smo isti govor posneli preko več sočasnih prenosnih kanalov, kar nam omogoča analize vplivov kanala na istem izvornem govornem signalu.

Govorci so pod nazorom operaterja snemanja govorili na tri načine: spontani govor, pogovor in branje.

Vsak način govora smo posneli v dolžini najmanj dveh minut. Vsako snemanje smo pričeli z branjem teksta nekega članka, pri čemer se je govorec lahko vsaj približno privadil na snemalne naprave. Po branju smo v pogovoru, ki smo ga snemali, govorca pripravili na ustrezno temo, ki mu je blizu. Pri tem je bil na posnetkih slišen tudi govor sogovornika, ki je vodil pogovor in snemanje. Na ta način smo skušali zagotoviti čim bolj naraven in sproščen način govora. V nadaljevanju smo posneli še spontani govor, ki je v obliki monologa o določeni temi trajal prav tako 2 minuti. Izkazalo se je, da govorcu močno olajšamo spontani govor v obliki monologa, če se le ta smiselno in tematsko navezuje na pričeti pogovor v prejšnjem delu snemanja, saj se ljudje praviloma počutijo nelagodno, ko morajo pred določeno osebo več časa nepripravljeni govoriti o poljubni temi.

Določeno težavo pri snemanju je predstavljala nesproščenost govorcev pri spontanem govoru. Izkazalo se je, da za določene ljudi predstavlja nelagodje, če jih na snemanje vnaprej ne pripravimo. Priprava je običajno obsegala obrazložitev postopka in namena snemanja. Nekateri govorci so želeli, da jim zagotovimo anonimnost oziroma zagotovilo, da posnetki ne bodo zlorabljeni ali javno objavljeni.

4. EKSPERIMENT

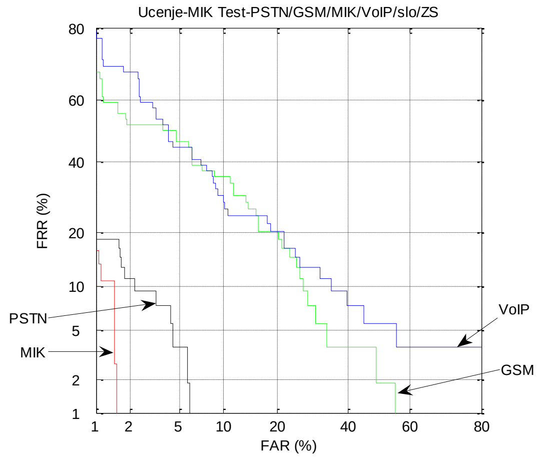
Meritev uspešnosti sistema za SRG [9] smo izvajali v več korakih: izbor posnetkov, generiranje modela ozadja, učenje sistema, testiranje in analiza rezultatov. Najprej smo izbrali dve skupini posnetkov. Prva skupina posnetkov je bila namenjena za generiranjem modela ozadja, druga skupina pa je bila razdeljena na podskupino za učenje in podskupino za testiranje sistema.

Pomembno je, da za **model ozadja** izberemo posnetke iz iste populacije, kot je zastopana v posnetkih za testiranje in učenje. Če so za model ozadja uporabljeni posnetki govorcev, ki hkrati govorijo tudi na posnetkih za testiranje in učenje, govorimo o testiranju v zaprtem podatkovnem setu. Model ozadja je pri vseh meritvah enak in je zgrajen iz dveh skupin posnetkov. Prva skupina vsebuje 66 posnetkov prek mikrofona, druga skupina pa 124 posnetkov prek mobilne telefonije. Za izgradnjo slovenskega modela ozadja smo morali zaradi omejene količine slovenskih govornih posnetkov vključiti tudi tiste posnetke, na katerih govorijo iste osebe kot na posnetkih za učenje in testiranje. Obe skupini posnetkov za model ozadja sta bil sestavljeni iz posnetkov pogovora in spontanega govora. Skupini mikrofonskih posnetkov smo torej dodali še 30 posnetkov moških govorcev iz javnih TV oddaj in parlamenta. Skupini posnetkov prek mobilne telefonije pa smo priložili 87 posnetkov pogovorov slovenskih moških. V teh posnetkih govori približno 10 različnih oseb, ki so posnete prek prisluhov v mobilnem telefonskem omrežju.

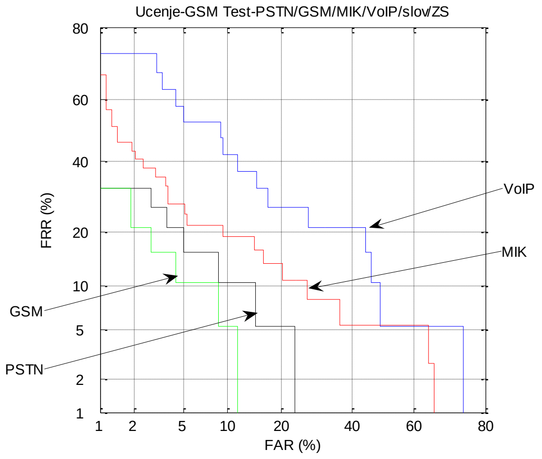
Učenje modelov govorcev je potekalo s posnetki, posnetimi preko vseh petih kanalov. Učenje s posnetki prek mikrofona, smo izvajali s posnetki prek obeh mikrofonov pri branju in spontanem govoru, torej skupno s štirimi posnetki za vsakega govorca. Sistem smo učili tudi s posnetki prek telefonije GSM, PSTN in VoIP za vseh izbranih 18 govorcev iz lastne govorne zbirke (branje in spontani govor).

Testiranje smo izvajali s posnetki iz lastne govorne zbirke prek vseh petih kanalov, ki so bili posneti pri pogovoru.

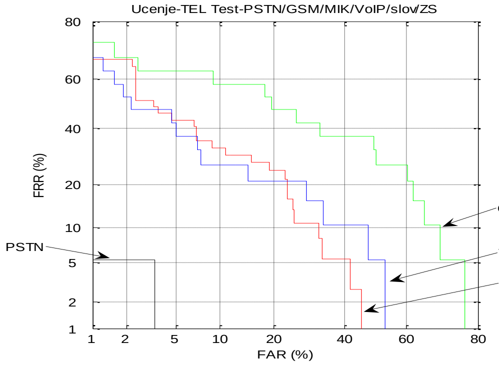
Za učenje smo uporabljali eno vrsto kanala, testiranje pa smo izvedli na podatkih posnetih prek istega in vseh preostalih kanalov. Tako smo lahko opazovali obnašanje sistema pri istem modelu ozadja, vendar pri različnih kanalih za učenje in testiranje. Na slikah 2, 3, 4 in 5 so prikazani rezultati štirih sklopov meritev uspešnosti sistema za SRG s posnetki za učenje modelov moških slovenskih govorcev, ki smo jih posneli prek štirih kanalov. DET krivulje uspešnosti sistema so na vseh grafih za mikrofonske posnetke obarvane rdeče, za PSTN posnetke črno, za GSM posnetke zeleno in za VoIP posnetke modro.



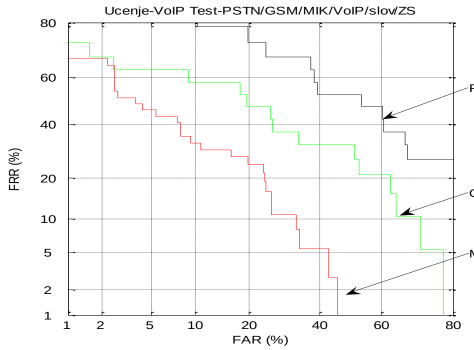
Slika 3. DET krivule učenja modelov z MIK posnetki.



Slika 4. DET krivulje učenja modelov z GSM posnetki.



Slika 5. DET krivulje učenja modelov z PSTN posnetki.



Slika 6. DET krivulje učenja modelov z VoIP posnetki.

5. REZULTATI

Iz rezultatov meritev uspešnosti sistema za SRG v mešanih okoliščinah lahko ugotovimo, da se sistem za SRG pričakovano najbolje obnaša s posnetki, pridobljenimi v istih razmerah tako za učenje kot testiranje.

FRR (angl. False Rejection Rate) je verjetnost, da sistem za SRG ne zazna govorca na posnetku, kjer je govorec prisoten. Govorimo o deležu napačno zavrnjenih govorcev.

FAR (angl. False Acceptance Rate) predstavlja verjetnost, da bo sistem za SRG napačno zaznal govorca, ki ni prisoten v posnetku, ki ga sistem analizira. Pri identifikaciji bo sistem identificiral govorca, ki ni prisoten v tesni množici, pri verifikaciji pa bo sistem napačno potrdil istovetnost neavtentičnega posnetka.

EER (angl. Equal Error Rate) predstavlja točko, kjer je verjetnost za napačno sprejetje in napačno zavrnitev enaka; torej je odločitev enaka naključnemu odločanju. Nižja kot je vrednost EER, boljši je sistem.

Pri učenju z mikrofonskimi posnetki (slika 3) dosega EER pod 5 % za mikrofonske testne posnetke. Obnašanje sistema s PSTN posnetki je nekoliko slabše, najslabše pa se sistem obnaša z GSM in VoIP posnetki, kjer je EER okoli 20 %.

Pri učenju s posnetki pridobljenimi preko mobilne telefonije GSM (slika 4) dosežemo EER okoli 10 % v primeru GSM tesnih posnetkov, nekaj nad 10 % v primeru PSTN posnetkov, 15 % v primeru uporabe mikrofona in nad 20 % pri uporabi testnih posnetkov preko VOIP.

V primeru učenja sistema za SRG s posnetki preko telefonije PSTN (slika 5) najboljši rezultat zasledimo pri testnih posnetkih, ki so posneti v enakih razmerah kot za učenje. Občutno slabše pa se sistem obnaša pri VoIP in mikrofonskih testnih posnetkih z EER okoli 20%. Testni posnetki preko GSM pa prinesejo napako EER blizu 40% .

Tudi v primeru učenja z VOIP posnetki (slika 6) so rezultati najboljši v primeru, ko so učni in testni posnetki pridobljeni na enak način. Opazimo pa lahko občutno poslabšanje napake pri posnetkih pridobljenih z mikrofonom in preko GSM telefonije z EER okoli 40%. Pri posnetkih pridobljenih s PSTN telefonijo znaša EER okoli 50%, kar pa je enakovredno naključnemu odločanju.

Iz vseh meritev v mešanih okoliščinah lahko ugotovimo, da se sistem za SRG pričakovano najbolje obnaša s posnetki, pridobljenimi v istih učnih in testnih razmerah. Vrednost EER za učne in testne posnetke pridobljene v enakih okoliščinah je razmeroma majhna, pod 5%, razen pri GSM, kjer je okoli 10%, pri čemer je pri VoIP celo pod 1%, tako da modra krivulja niti ni več vidna na grafu. To pripisujemo razmeroma majhni testni zbirki posnetkov. Gre namreč za testiranje v zaprtem podatkovnem setu.

6. ZAKLJUČEK

Prdstavili smo problematiko razpoznavanja oz. verifikacije govorcev v forenzične namene. Poudarek je bil na problematika pridobivanja posnetkov pod »stvarnimi pogoji«. V zvezi s tem smo proučevali vpliv različnih načinov zajemanja govornega signala na rezultate prepoznavanja SRG. Za potrebe eksperimenta je bila posneta posebna govorna zbirka slovenskih govorcev, ki jo je smiselno dograjevati z novimi glasovi.

Izkazalo se je, da so sistemi za SRG še vedno precej občutljivi na vplive prenosnega kanala. Največji izzivi so v primerih, ko izvajamo učenje sistema na podatkih, ki so pridobljeni preko ene vrste telefonije, testiranje sistema pa se izvaja na podatkih, ki so posneti preko druge vrste telefonije oziroma neposredno preko mikrofona. Izkazalo se je, da so rezultati v primerih mešanih pogojev znatno slabši od rezultatov pridobljenih v enakih pogojih.

7. LITERATURA IN VIRI

[1] Ortega-Garcia, J., Gonzalez-Roidriguez, J., Marrero-Aguiar, V., 2000. AHUMADA: A large speech corpus in Spanish for speaker characterization and identification, *Speech Communication 31*, str. 255-264.

[2] Šef, T., Baucon, P., 2007. Sodno izvedenstvo in razpoznavanje (identifikacija) govorcev v kazenskem postopku, *Pravosodni bilten*, 2/2007.

[3] Hollien, H.,2002. *Forensic Voice Identification*, Academic Press.

[4] Rose, P., 2005. Technical forensic speaker recognition: evaluation, types and testing of evidence, *Computer Speech and Language*.

[5] Alexander, A., 2005. *Forensic Automatic Speaker Recognition Using Bayesian Interpretation and Statistical Compensation for Mismatched Conditions*, doktorska disertacija, Lausanne, EPFL.

[6] Rose, P., 2002. *Forensic speaker Identification*, Taylor & Francis.

[7] Jessen, M., 2007. *Some Experiences from Tests of an Automatic Speaker Recognition System under Forensic Conditions*, Bundeskriminalamt, EAFS.

[8] Blatnik, R., 2012. *Vpliv kakovosti govora v telefoniji na samodejno razpoznavanje govorca*, magistrska naloga.

[9] http://www.persay.com/pdf/SPID_V6_DataSheet.pdf

JSI Sound – platforma za enostavno klasifikacijo zvočnih posnetkov: Demonstracija na zvokih živali

Borut Budna, Martin Gjoreski,
Anton Gradišek, Matjaž Gams
Odsek za inteligentne sisteme,
Institut "Jožef Stefan"
Jamova cesta 39, SI-1000 Ljubljana
anton.gradisek@ijs.si

POVZETEK

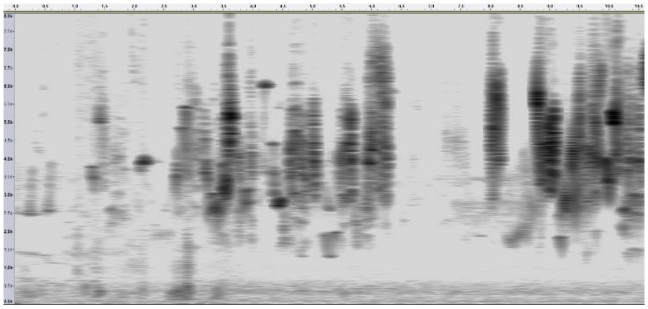
Predstavljamo orodje JSI Sound, ki je namenjeno enostavni klasifikaciji zvočnih posnetkov. Implementirano je v okolju Orange, ki je odprtokodno orodje za strojno učenje in vizualizacijo podatkov za strokovnjake in začetnike. JSI Sound je bil razvit v skladu s paradigmo »strojno učenje kot storitev«, saj omogoča enostavno testiranje klasifikacijskih modelov na različnih bazah podatkov zvočnih posnetkov, v mislih imamo predvsem različne biozvoke. S tem je primeren tako za ljubitelje s področja bioakustike brez naprednega znanja s področja strojnega učenja, ki lahko JSI Sound uporabijo kot enostavno klasifikacijsko orodje, kot tudi za strokovnjake, ki ga lahko uporabijo za enostavno testiranje modelov kot prvi korak pri izdelavi specializiranih klasifikacijskih aplikacij. Vhodne podatke za JSI Sound predstavlja serija označenih posnetkov. Uporabnik v seriji korakov s pomočjo grafičnega vmesnika izbere način filtriranja, segmentacije in postopek določitev značilk. Na podlagi teh značilk orodje zgradi serijo klasifikacijskih modelov in jih testira. Tu predstavimo testiranje sistema na treh serijah podatkov – na brenčanju čmrljev ter oglašanju ptic in žab.

Ključne besede

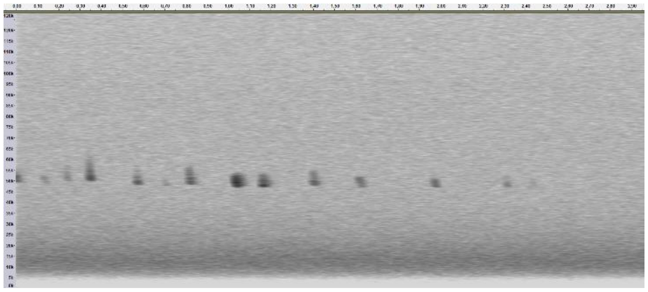
Živalsko oglašanje, strojno učenje, Orange, klasifikacija

1. UVOD

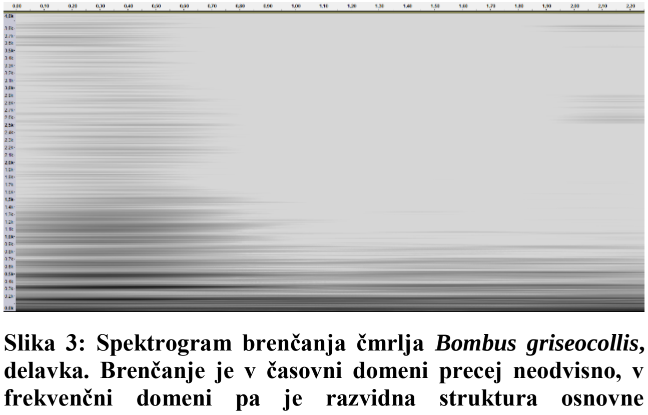
Metode umetne inteligence in strojnega učenja so dolgo temeljile na analizah strogo strukturiranih podatkov, v novejšem času pa se vedno bolj ukvarjajo z direktnimi podatki iz realnega sveta, kot so video in avdio posnetki. V tem prispevku se osredotočimo na analizo in klasifikacijo zvokov, ki jih proizvajajo živali. Naloga je pomembna v zoologiji, denimo v študijah biotske raznolikosti. Čeprav je mnoge živalske vrste enostavno prepoznati na podlagi videza, to ni vedno možno – bodisi zaradi življenjskega sloga (mnoge ptice se denimo skrivajo v grmovju ali v trsju) bodisi zaradi tega, ker so si osebki več vrst tako podobni, da jih lahko ločimo šele ob podrobnem morfološkem pregledu – to pa zahteva, da osebek ujamemo. Klasičen primer so penice, skupina ptic, ki so si na videz precej podobne, se pa vsaka vrsta izrazito drugače oglaša, kar lahko uporabimo kot osnovo za prepoznavanje. Drug primer so netopirji – med letom ponoči je vizualno prepoznavanje izredno zahtevno ali celo nemogoče, lahko pa jih prepoznavamo na podlagi oglašanja – netopirji se oglašajo v ultrazvočnem območju, zvoke pa uporabljajo za eholokacijo in za sporazumevanje. Še en primer so žuželke na travniku, te proizvajajo različne vrste zvokov, kot so klici za sporazumevanje ali zvok brenčanja med letom. Slike 1-3 prikazujejo primere spektrogramov za oglašanje ptice, netopirja in čmrlja. Vidimo, da so si izrazito različni med seboj, tako v časovni kot v frekvenčni domeni.



Slika 1: Spektrogram oglašanja ptice *Sylvia communis* (rjava penica). Iz spektrograma je razvidno izrazito strukturirano oglašanje, tako v časovni kot tudi v frekvenčni domeni.



Slika 2: Spektrogram klicev naključnega netopirja. Gre za kratke eholokacijske klice v ultrazvočnem območju.



Slika 3: Spektrogram brenčanja čmrlja *Bombus griseocollis,* delavka. Brenčanje je v časovni domeni precej neodvisno, v frekvenčni domeni pa je razvidna struktura osnovne frekvence in višje harmonskih frekvenc.

Problem klasifikacije živalskih vrst z metodami strojnega učenja na podlagi oglašanja ni nov – v literaturi zadnjih let najdemo primere za različne skupine živali in za različne pristope. Gradišek et al. [1] so uporabili kombinacijo metod strojnega učenja za prepoznavanje nestrukturiranega brenčanja čmrljev, kot najboljša metoda se je izkazal naključni gozd. Ganchev in Potamitis [2] sta se ukvarjala z oglašanjem žuželk (črički, škržati, prave cvrčalke), uporabila sta kombinacijo probabilističnih nevronskih mrež in gaussovskih modelov. Dosegla sta 90 % klasifikacijsko točnost na bazi 307 vrst. Stowell in Plumbey [3] sta uporabila nenadzorovano učenje za prepoznavanje ptic, različne metode za prepoznavanje ptic so preizkušali tudi Cheng et al. [4].

Kot smo videli na primerih spektrogramov, ima oglašanje vsake od skupin živali svoje posebnosti, zato se ni enostavno odločiti, na kakšen način bomo pristopili h klasifikacijskemu problemu – kakšne značilke izbrati in katere algoritme uporabiti. Odločitev je še težja za strokovnjake s področja bioakustike, ki bi si želeli delujoče klasifikacijske aplikacije, nimajo pa obširnega znanja s področja strojnega učenja. Naša rešitev JSI Sound [5] izhaja iz paradigme »strojno učenje kot storitev«. Izdelali smo orodje, s katerim lahko uporabnik s pomočjo grafičnega vmesnika preizkusi različne pristope pri gradnji klasifikacijskih modelov. Po eni strani je to lahko že dovolj za enostavne klasifikacije, po drugi strani pa predstavlja dobro osnovo za gradnjo robustnih specializiranih aplikacij. Naše orodje se razlikuje od obstoječih rešitev, ki jih lahko razdelimo predvsem na orodja v obliki knjižnic ali modulov (npr. pyAudioAnalysis [6]) ter orodja z grafičnim vmesnikom (kot je denimo Audacity [7]). Primerjava nekaterih funkcij za vsako od orodij je prikazana v Tabeli 1.

Tabela 1: Primerjava funkcij treh različnih orodij za obdelavo zvoka

	Audacity	**pyAudioAnalysis**	**JSI Sound**
Spektrogram	da	da	da
Filtriranje	da	ne	da
Segmentacija	da (ročna)	da	da
Ekstrakcija značilk	ne	da	da
Klasifikacija	ne	da	da
GUI	da	ne	da

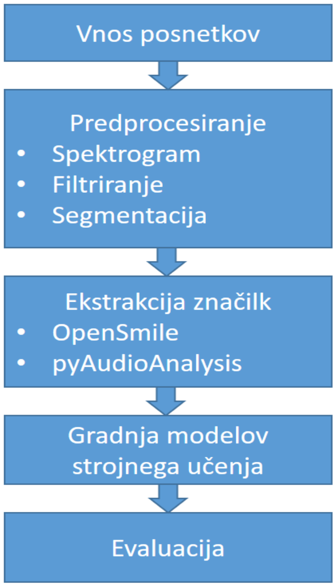
Orodje JSI sound je razvito v okolju Orange [8,9], s tem je prosto dostopno. V prispevku opišemo splošno delovanje metode, implementacijo v okolju Orange in rezultate testiranja na treh skupinah posnetkov živalskih zvokov.

2. METODA

Splošna metoda je prikazana na Sliki 4. Sestavljena je iz petih korakov: vnos zvočnih posnetkov, predprocesiranje, ekstrakcija značilk, gradnja modelov strojnega učenja ter evaluacija modelov.

Vhodne podatke predstavlja skupina označenih posnetkov, zaželeno je, da so bili vsi pridobljeni z enako opremo in z enako frekvenco vzorčenja. Predprocesiranje zajema uporabo filtrov za odstranitev šuma. Izberemo lahko med petimi filtri; FIR, Butterworth, Čebišev, Eliptični in Besselov filter. V tem koraku izvedemo tudi segmentacijo na posnetke izbrane dolžine. Uporabimo lahko fiksno ali drseče okno. Hkrati lahko zavržemo segmente, v katerih ni dovolj informacije.

Za ekstrakcijo značilk uporabimo odprtokodne knjižnice, trenutno uporabljamo OpenSmile [10] in pyAudioAnalysis [6]. Te knjižnice podpirajo veliko število značilk (ki so rezultati kompleksnih matematičnih operacij na vhodnih posnetkih), tako v časovni kot tudi v frekvenčni domeni. Značilke temeljijo na operacijah, kot so mel-frekvenčni kepstralni koeficienti (MFCC), koeficientih percepcijskega linearnega napovedovanja (PLP) in koeficientih Chroma [10]. Ta pristop se je že izkazal koristnega pri analizi človeškega govora [11].



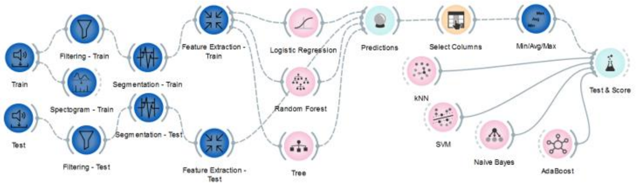
Slika 4: Shema metode, ki jo uporablja orodje JSI Sound

Na podlagi značilk Orange zgradi odločitvene modele, kot so naključni gozd, SVM, naivni Bayes in druge. Za evaluacijo posameznega modela se podatke razdeli na učno množico, na kateri se trenira modele, in na testno množico, na kateri se te modele nato testira. Pri tem JSI Sound skrbi, da so vsi segmenti, ki pripadajo istemu začetnemu posnetku, vedno ali v učni ali v testni množici. Poleg tega omogoča gradnjo modelov na nivoju posameznega segmenta ali pa na nivoju celotnega posnetka, z uporabo kombinacije napovedi za vsakega od segmentov posebej.

3. IMPLEMENTACIJA V OKOLJU ORANGE

Po namestitvi je vtičnik JSI Sound dostopen v klasičnem seznamu vtičnikov okolja Orange. Uporabnik naloži bazo posnetkov, nato s seznama izbere ustrezne filtre ter parametre za segmentacijo posnetkov (dolžina in prekrivanje okna). Zatem uporabnik izbere knjižnice za ekstrakcijo značilk.

V naslednjem koraku uporabnik s pomočjo orodij v okolju Orange gradi modele ter jih evalvira. Primer uporabe JSI Sound je prikazan na Sliki 5.



Slika 5: Primer uporabe orodja JSI Sound v okolju Orange. Vtičniki, ki so pobarvani modro, so bili razviti za JSI Sound, ostali so standardni vtičniki okolja Orange.

Po opisu implementacije in sheme uporabe lahko uporabnik vidi, da je orodje enostavno za uporabo, poleg tega pa mu omogoča hitro in natančno gradnjo klasifikacijskih modelov, katerih natančnost je primerljiva z rezultati prikazanimi v Tabeli 3.

4. EKSPERIMENTI

Orodje JSI Sound smo testirali na treh različnih skupinah živalskih zvokov, na posnetkih oglašanja slovenskih žab in ptic iz družine penic ter na posnetkih brenčanja čmrljev, gre za vrste iz zvezne države Kolorado v ZDA. Število razredov, posnetkov in segmentov za vsako od skupin prikazuje Tabela 2.

Tabela 2: Struktura podatkov za vsako od skupin živali

	Ptice	**Žabe**	**Čmrlji**
Št. razredov	6	13	9
Št. posnetkov	81	39	51
Št. segmentov	5536	4447	3854

Eksperiment je sestavljen iz treh korakov: strojnega učenja na nivoju segmentov, ekstrakcije značilk na nivoju posnetkov in strojnega učenja na nivoju posnetkov. Motivacija za ta pristop je dvojna: ker vemo, da posamezni segmenti istega posnetka pripadajo istemu razredu, nam kombinacija informacij o več segmentih lahko pove več kot le informacija o posameznem posnetku. Več metod strojnega učenja pa kombiniramo zato, ker lahko različne metode delujejo različno dobro na posameznih strukturah v podatkih.

Za strojno učenje na podlagi posnetkov smo uporabili sledeče metode: Logistična regresija (LR), Naivni Bayes (NB), metoda najbližjega soseda (kNN), naključni gozd (RandomForest, RF) in AdaBoost. Vhodni podatek za vsako od metod je vektor značilk za vsakega od segmentov, izhodni podatek pa so verjetnosti za vsakega od klasifikacijskih razredov.

V koraku ekstrakcije značilk na podlagi posnetkov združimo napovedi modelov iz predhodnega koraka, uporabimo maksimalno, minimalno in povprečno vrednost napovedi vsakega od modelov.

V koraku strojnega učenja na osnovi celotnih posnetkov na podlagi napovedi posameznih modelov na nivoju segmentov izdelamo meta-klasifikatorje. Te preverimo s desetkratnim prečnim preverjanjem na testni množici. Rezultati, v metriki ploščine pod krivuljo (area under curve, AUC), so predstavljeni v Tabeli 3.

Tabela 3: Klasifikacijski rezultati v metriki AUC z 10-kratnim prečnim preverjanjem za vsakega od modelov za vse tri skupine živali

	Ptice	**Žabe**	**Čmrlji**
LR	**94**	**100**	80
kNN	92	99	75
RF	**94**	100	**81**
NB	89	99	74
AdaBoost	85	93	67

Rezultati za ptice so odlični in bodo osnova za izdelavo namenske aplikacije. Visoka klasifikacijska točnost za žabe je verjetnost posledica majhnega števila posnetkov v vsakem od razredov, za izboljšanje zanesljivosti bo potrebnih več posnetkov. Rezultati za čmrlje so blizu tistim, ki smo jih dobili na bazi posnetkov slovenskih vrst [1].

5. ZAKLJUČEK

Predstavljamo orodje JSI Sound, ki je bilo razvito z namenom olajšati naloge s področja strojnega učenja za uporabnike, ki nimajo naprednih izkušenj s tega področja ali s področja obdelave zvočnih posnetkov. Orodje JSI Sound je implementirano v okolju Orange in vsebuje pet metod filtriranja, dve metodi segmentacije posnetkov ter dve obsežni knjižnici za določanje značilk, tako v časovni kot tudi v frekvenčni domeni. Vse te funkcije so dostopne kot vtičniki za Orange, do njih pa dostopamo prek grafičnega vmesnika.

Sistem smo testirali na treh setih posnetkov živalskih zvokov – na posnetkih brenčanja čmrljev ter oglašanju ptic iz družine penic ter žab. Rezultati klasifikacijskih modelov so podobni tistim, ki smo jih na istih ali podobnih setih dobili v prejšnjih študijah, kar kaže na primernost orodja JSI Sound za tovrstne naloge. V teku je raziskovanje primernosti metode za analizo drugih tipov zvokov, denimo za analizo govora in zvokov človeškega telesa, kot sta bitje srca in dihanje.

6. ZAHVALA

Zahvaljujemo se dr. Tomiji Trilarju za posnetke ptic in žab, Primožu Presetniku za posnetek netopirja ter prof. Candace Galen in dr. Nicole Miller-Struttmann za posnetke čmrljev.

7. VIRI

[1] Gradišek, Anton, Gašper Slapničar, Jure Šorn, Mitja Luštrek, Matjaž Gams, and Janez Grad. 2017. 'Predicting species identity of bumblebees through analysis of flight buzzing sounds', Bioacoustics, 26: 63-76.

[2] Ganchev, Todor, and Ilyas Potamitis. 2007. 'Automatic acoustic identification of singing insects', Bioacoustics, 16: 281-328.

[3] Stowell, Dan, and Mark D Plumbley. 2014. 'Automatic large-scale classification of bird sounds is strongly improved by unsupervised feature learning', PeerJ, 2: e488.

[4] Cheng, Jinkui, Bengui Xie, Congtian Lin, and Liqiang Ji. 2012. 'A comparative study in birds: call-type-independent species and individual recognition using four machine-learning methods and two acoustic features', Bioacoustics, 21: 157-71.

[5] Budna, Borut. 2017. 'Platform for audio clips classification', University of Ljubljana, Faculty of Computer and Information Science.

[6] https://github.com/tyiannak/pyAudioAnalysis

[7] http://www.audacityteam.org/

[8] Demšar, Janez, Blaž Zupan, Gregor Leban, and Tomaz Curk. 2004. "Orange: From experimental machine learning to interactive data mining." In European Conference on Principles of Data Mining and Knowledge Discovery, 537-39. Springer.

[9] Demšar, Janez, Tomaz Curk, Aleš Erjavec, Črt Gorup, Tomaž Hočevar, Mitar Milutinovič, Martin Možina, Matija Polajnar, Marko Toplak, and Anže Starič. 2013. 'Orange: data mining toolbox in Python', Journal of Machine Learning Research, 14: 2349-53.

[10] Eyben, Florian, 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, 1459-62. ACM

[11] Martin Gjoreski, Hristijan, Gjoreski, and Andrea Kulakov. 2014. Machine learning approach for emotion recognition in speech. Informatica, vol. 38, no. 4, pp. 377-384.

Bat Classification using Deep Neural Network

Jani Bizjak
Department of Intelligent Systems,
Jožef Stefan Institute
Jamova cesta 39
1000 Ljubljana, Slovenia
+386 1 477 3147
jani.bizjak@ijs.si

Anton Gradišek
Department of Intelligent Systems,
Jožef Stefan Institute
Jamova cesta 39
1000 Ljubljana, Slovenia
+386 1 477 3147
anton.gradisek@ijs.si

Luka Stepančič
Department of Intelligent Systems,
Jožef Stefan Institute
Jamova cesta 39
1000 Ljubljana, Slovenia
+386 1 477 3147
luka.stepancic@ijs.si

Primož Presetnik
Center za Kartografijo Favne in Flore
Klunova 3
1000 Ljubljana, Slovenia
primoz.presetnik@ckff.si

ABSTRACT

We present a deep neural network approach for bat species classification by echolocation and social calls. First the data is gathered on two separate locations using special high frequency ultrasound recorder. The data is then preprocessed in order to be usable in deep neural network architecture. Deep architecture used is discussed and experimental results for classification of two species are presented. The last part of the paper focuses on future work that could improve results.

General Terms

Algorithms, Design, Experimentation

Keywords

bats, deep neural network, convolutional neural networks, Pipistrellus pygmaeus, Barbastella barbastellus

1. INTRODUCTION

Bats are second largest order of mammals, representing 20% of all mammal species worldwide. Bats live in most of the world except extremely cold regions. There are 30 species of bats classified in Slovenia out of more than 1200 found around the world.

Bats in Slovenia are all insectivores – meaning they feed on insects. They are mostly active during the dusk and in general have poorly developed vision. For navigation and hunting however they use special organ that works similarly to sonar. They emit ultrasonic sounds, between 10 and 120 kHz and are precise enough to detect less than a 0.01 mm wide obstacle [1].

They live in colonies and also have ultrasonic social calls in order to communicate with each other. The sounds they produce differs from specie to specie and can be used for classification.

Bats perform vital ecological roles of pollinating flowers, they consume insect pests and their excrements (guano) are very good fertilizers. It is thus important for humans to do what they can to help them prosper. In this paper we focus on two species found in Slovenia, *Barbastella barbatellus* and *Pipistrellus pygmaeus* [Figure 1].

Modern ICT solutions nowadays allow us to better understand and study these animals by using automated systems. Since bats use such a unique form of communication they can be easily detected using sound compared to other methods (image), however sound data is full with noise and it might not be an easy task to differ between two species of bats. Visual recognition can often be a difficult task when dealing with lowlight conditions, which are usually present when bats are normally active. On the other hand, classification based on bat calls is a more promising approach. Experts can classify most of Slovenian bat species based on the audio recordings. However, automatic classification methods are desired since they allow us to process large amounts of data from recording stations.

In last years deep neural networks (DNN) proved to perform very well in classification of real world signals such as sounds or images [2]. In this paper we present an advanced deep learning architecture that can be used for bat species classification based on sound.

Barbastella barbatellus *Pipistrellus pygmaeus*

Figure 1: Two species of bats found in Slovenia that we try to classify.

2. Data

Data consist of two datasets from two different location Dolenja Vas; around 2000 recordings, 8 GB in size, and Kozina which contains 8.900 recordings and is 28 GB in size.

The data were recorded using a specialized high frequency recorder SM4BAT FS&ZC. Recorder has a recording range of 16 Hz to 150 kHz and recording frequency of 500 kHz. We used band pass filter between 10 kHz to 110 kHz in order to filter out most of the noise and to lower amount of empty recordings (in general bats do not produce noise that is lower than 15 kHz or higher than 100 kHz at the same time there are not many animals that produce sounds in such frequency range).

The recorder automatically records a sound, couple of seconds earlier and couple of seconds after certain frequency threshold is reached e.g. when sound over 15kHz is detected. The recording is then manually labeled by the expert, who normally looks at the spectrogram in order to determine number and species of bats in the recording. In total the dataset contains 22 different species. Majority of cases consists of one specie per recording although some contain multiple species. In some recordings expert was unable to determine exact specie so a specie family is used as a label. In total there are 37 labels in this dataset.

2.1 Data preprocessing

The information in wave form is concentrated in certain frequency components which are impossible to detect in wave form format that is why Furrier transform is performed to transform data into time-frequency representation – spectrogram. Spectrogram format also gives us a better look into frequency distribution in a signal (it is easier for human to interpret a visual information). The frequency resolution of spectrogram used was 256.

A representative spectrogram is shown in Figure 2. It is easy to distinguish 3 types of signals. Below 20 kHz there is noise, which can be discarded. At higher frequencies, there are two types of patterns. A social call, which spans over a higher range of frequencies, and an echolocation call, that is used for navigation.

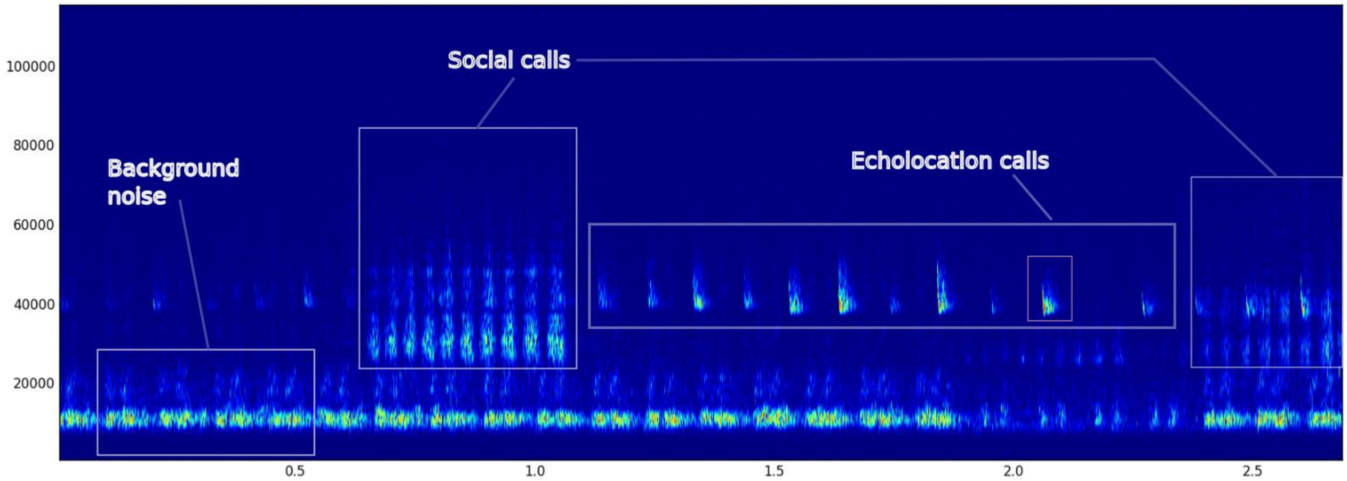


Figure 2: Spectrogram of one recording. In the figure 3 distinct patterns emerge. Everything below 20 kHz is noise, in the range of 20kHz and 100 kHz there are two type of sounds, social calls and echolocation calls.

Due to the high sampling frequency (500kHz) of the recording device it is not possible to feed whole signal to the neural network. We split the recording using a sliding window of size 2048 samples. To reduce the size of the file we also removed frequencies below 15 kHz and above 80 kHz, which did not contain any relevant information for classification of the two bat species that we attempted to identify.

When dealing with natural signals such as sound or image, it is best to feed the neural networks with raw signal and allow for their abstraction power to generalize information out of the data. Spectrogram can be represented to the neural network as an image - that is why we do not attempt to manually extract any more features but feed the windows extracted to the network.

3. DEEP ARCHITECTURE

Deep neural networks are becoming increasingly popular in the last years. They perform especially well on natural signals and are dominant on the domains of image recognition, voice recognition and natural language recognition. In our experiments we tested several deep architectures that we present in the following sections.

3.1 Deep neural network

Artificial neural networks have existed since 1954 when Farley and Weasley first implemented a simple neural network on a computer. Because of the computation complexity they were not used widely until the late 2000's, when computers with new architectural design, initially optimized for graphic cards to allow parallelization, became capable enough to run several layers of neurons – a deep architecture. Architecture is considered deep if it contains at least 4 layers of neurons.

In 2012, AlexNet [3] architecture was proposed for image recognition. The network achieved more than a 10 % increase in accuracy compared to the second best method and made DNNs one of the most used ML methods today. The architecture uses convolutional layers to generalize input image and combine previously learned features into more complex high level features. By the rule of thumb, Convolutional Neural Networks (CNN) can be used and often perform good, especially when the task can be presented in such a way that a person can use their bare eyes to classify the objects. Since spectrograms are analyzed by humans by looking at them and discovering patterns in the time-frequency representation it is likely that CNN will perform better than other architectures on the same domain.

We loosely based our architecture on several state-of-the-art architectures [3,4,6] and fine tune it to achieve best results for bat domain. In our architecture we used 3 convolutional layers, each consisting of 34 filters sized 7x7, 5x6, and 3x3 on each consecutive layer. After the second and third layer, we use MaxPooling layer which uses filter size 2x4 and 2x2. Max pooling is used to reduce size of the image (data). Filter slides over the image and concatenates all values under the filter into one. Different approaches can be used e.g. average, min, or max [4].

After the fifth layer, 3 fully-connected layers follow. In order to reduce over-fitting, a small number of neurons is used 8, 4, and 2 on layers 6, 7, 8. We use rectified linear activation function (ReLU) [Figure 3] which in general gives best performance [5].

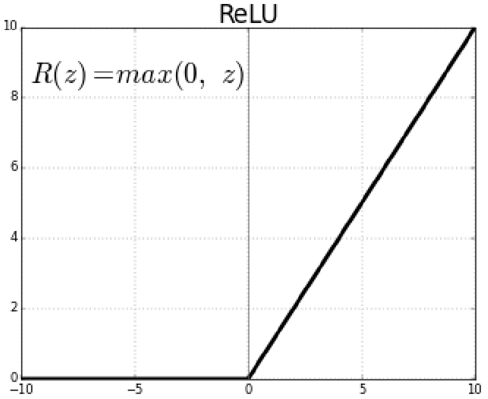


Figure 3: Rectifier Linear Function gives good performance and is extremely easy to calculate on GPU where float operations are computational expensive (example sigmoid)

In the last layer we use SoftMax regression for classification. The whole architecture is presented in Figure 4.

In order to speed up the learning, instead of using standard gradient descend (SGD) for adjusting neuron weights we use RMSProp [4] which supports batched learning - allows for parallelization, adjust gradient weight for each parameter separately and uses adaptive normalization with decay parameter β [Equation 1].

$$v = \beta v + (1 - \beta)(\Delta f)^2$$

$$\theta = \theta - \alpha \frac{\Delta f}{\sqrt{v} + \varepsilon}$$

Equation 1 RMSProp

Despite large amounts of data, we discovered our network was still over-fitting. In order to resolve this problem, we introduced dropout [6] in the last layer. During training phase, the dropout method will randomly remove connections from neurons in one layer making the layer temporarily sparsely connected. The removed connections will be shuffled after each iteration preventing co-adaptations of neurons during learning. In the testing phase the model is again fully connected (dropout only works during learning phase). We used dropout probability of 0.5.

4. EXPERIMENTS

Recordings in the dataset last multiple seconds and contain several calls from one or multiple bats. Because of the large sampling frequency, it is not possible to input the whole recording into a neural network so we use windows. The problem with windows is that they are not labeled separately but have the same label as the original recording. This presents a problem with windows that do not contain any bat call. In order to alleviate this problem, we only focused on distinguishing between two species of bats *Barbastella barbatellus* and *Pipistrellus pygmaeus*. They were chosen because they had better noise to data ratio.

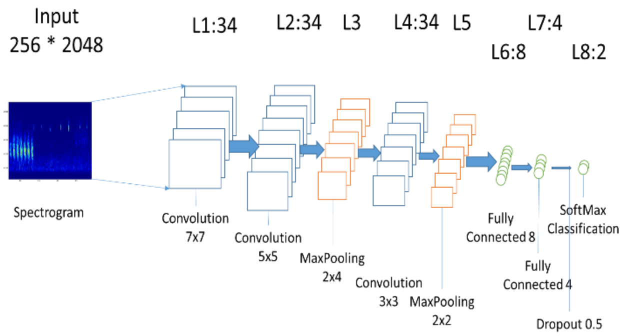


Figure 4: Network Architecture, uses 3 convolutional layers, 2 MaxPooling and 3 fully connected layers in the end.

Despite large amount of data in total there are only 128 recordings of the selected species. We split the recordings into 793 half second windows using sliding window technique. The amount of data available for training is extremely small for deep learning. Even worse is that a lot of windows from this data only contain noise and no bat calls.

We split the data 75% for training and 25% for testing and despite all achieved average 8% improvement compared to the majority class, which indicates that the networks were able to learn something from the data.

5. FUTURE WORK

Our initial experiments showed that deep neural networks have potential in bat classification based on sound. Currently the main problem is data segmentation and its labels. The one label per recording, which is then divided into multiple windows (with the same label) brings a lot of noise into the data. One solution would be to use whole recording as one instance of a class, but because of vast amount of data in one recording it is not currently feasible to do it. There are two solutions for this problem: have a better pre-processing and only learn on correctly labeled segments or introduce a new architecture of deep neural networks that can learn on unlabeled datasets.

For our next step we have implemented a segmentation method that uses power envelope to isolate bat calls from random noise or empty recordings as seen in Figure 5. Our initial tests show promising results (around 2 times better precision over majority

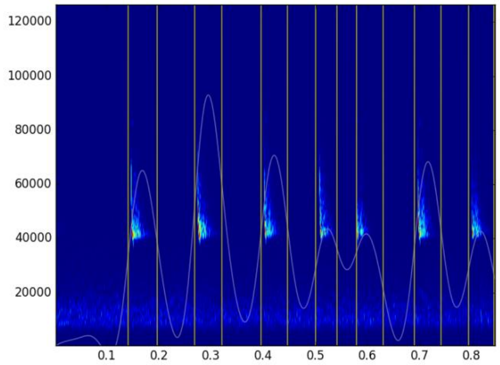


Figure 5: Bat calls segmentation using power envelope

class). However, it is likely that some of the time/sequential information is lost with this process.

In order to avoid this, the segmentation method can be used to first train the network to detect any bat sounds. When the network is proficient enough in the later task it can then be used to differentiate between different species. By dividing the problem into two sub problems the complexity/depth of the network can be lowered which allows for faster learning time and mitigates overfitting to an extent.

6. CONCLUSION

In this paper we presented problem of bat classification based on ultrasound recordings. We recorded an extensive database of bat recordings and presented a deep architecture used for species classification. In last part we presented initial results and proposed methods in future work to improve current model.

Despite the initial poor results, which we think are the result of noisy data, we have shown that deep neural networks can be used for animal sound classification – more precisely bats. We believe that with more data and better segmentation greater improvements in accuracy can be achieved. Additionally, semi-supervised learning can be used to train the model on vast amount of unlabeled data.

7. REFERENCES

[1] Slovensko Društvo za Proučevanje in Varstvo Netopirjev, http://www.sdpvn-drustvo.si/ (accessed 2017)

[2] Schmidhuber, Jürgen. "Deep learning in neural networks: An overview." Neural networks 61 (2015): 85-117..

[3] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems. 2012.

[4] Bizjak, Jani. Analiza signalov EKG z globokimi nevronskimi mrežami: magistrsko delo. Diss. J. Bizjak, 2015.

[5] Maas, Andrew L., Awni Y. Hannun, and Andrew Y. Ng. "Rectifier nonlinearities improve neural network acoustic models." Proc. ICML. Vol. 30. No. 1. 2013.

[6] Srivastava, Nitish, et al. "Dropout: a simple way to prevent neural networks from overfitting." Journal of machine learning research 15.1 (2014): 1929-1958.

Globoke nevronske mreže in matrična faktorizacija

Gašper Petelin
Univerza v Ljubljani, FRI
Večna pot 113
Ljubljana
gp6279@student.uni-lj.si

Igor Kononenko
Univerza v Ljubljani, FRI
Večna pot 113
Ljubljana
igor.kononenko@fri.uni-lj.si

POVZETEK

Predlagana je nova inicializacija globokih nevronskih mrež, ki pred začetkom učenja nevronske mreže nastavi uteži in pristranske vrednosti tako, da usmeri nevronsko mrežo proti rešitvi, ki pri optimizaciji hitreje konvergira k lokalnemu minimumu. Predlagana inicializacija je od običajne inicializacije precej počasnejša, vendar lahko pri globokih nevronskih mrežah precej pohitri skupni čas učenja.

Ključne besede

Globoke nevronske mreže, klasifikacijska točnost, regresija, nenegativna matrična faktorizacija, analiza arhetipov, inicializacija uteži.

1. UVOD

Globoke nevronske mreže v zadnjih letih dosegajo precej dobre rezultate [6][7], vendar se z večanjem števila skritih nivojev pojavijo tudi nekatere težave, kot so izguba gradienta pri vzvratnem širjenju napake in čas, ki je potreben za učenje. Predlagana je nova metoda za nastavljanje začetnih uteži, ki temelji na inicializaciji s pomočjo matrične faktorizacije, kjer je cilj, da bi pospešili učenje in preprečili preveliko izgubo gradienta.

2. PREDLAGANA INICIALIZACIJA

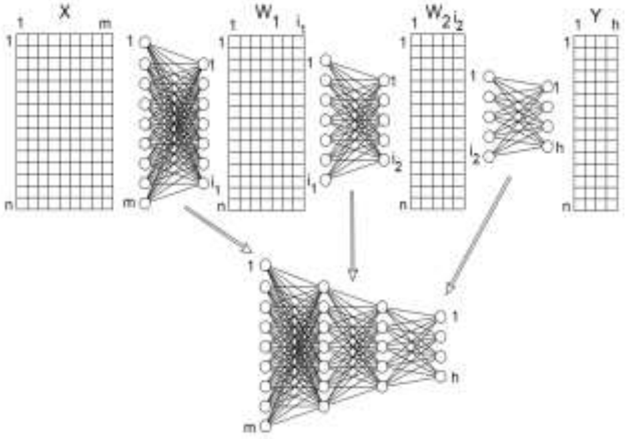
Ideja za predlagano inicializacijo delno izhaja iz inicializacij uteži z algoritmi za nenadzorovano učenje, kjer poskušamo vsak nivo nevronske mreže naučiti čim bolj abstraktno predstavitev podatkov, iz katerih lahko nato lažje pravilno napovemo vrednosti na izhodu. Pri predlaganem algoritmu te abstraktne predstavitve podatkov izračunamo s pomočjo matrične faktorizacije.

Predpostavimo, da imamo globoko nevronsko mrežo z k skritimi nivoji, pri kateri bi se radi naučili preslikati matriko podatkov $X \in \mathbb{R}^{n \times m}$ v izhodne podatke $Y \in \mathbb{R}^{n \times h}$. Konstante n, m in h predstavljajo število učnih primerkov, dimenzijo učnih podatkov in število nevronov na izhodu ciljne nevronske mreže. Originalno matriko X najprej faktoriziramo v pare $\{W_k \in \mathbb{R}^{n \times i}, H_k \in \mathbb{R}^{i \times m}\}$, kjer rank i predstavlja število nevronov ciljne nevronske mreže na skritem nivoju k. Število teh parov je zato enako številu nivojev v ciljni nevronski mreži. Dobljene matrike W v tem primeru predstavlja abstrakten povzetek originalne matrike oziroma aktivacije ciljne mreže na posameznem skritem nivoju. Matrik H za inicializacijo ne potrebujemo, zato jih lahko završemo.

Naslednji korak inicializacije je učenje k+1 enonivojskih mrež, kjer se vsaka od mrež nauči preslikavo iz ene predstavitve podatkov W_k v drugo predstavitev W_{k+1}. Posebnost so le prva enonivojska mreža, ki slika iz matrike X v W_1 in zadnja enonivojske mreža, ki slika iz W_k v matriko izhodnih podatkov Y. Za faktorizirane pare velja, da ni možno, da je dimenzija i večja od dimenzije m v matriki X oziroma velja $i_1, i_2, \ldots, i_k < m$.

Slika 1 prikazuje predlagano inicializacijo za ciljno mrežo z dvema skritima nivojema. Prvi korak je faktorizacija matrike X v dve novi matriki W_1 in W_2. Ko izračunamo abstraktno predstavitev podatkov za oba skrita nivoja, zgradimo tri enonivojske mreže, kjer prva mreža preslika originalno matriko X v prvo zgoščeno vrednost W_1, druga mreža slika iz W_2, zadnja mreža pa iz W_2 v končno matriko Y.

Za učenje enonivojskih mrež lahko uporabimo različne funkcije napak z različnimi aktivacijskimi funkcijami. Pri učenju enonivojskih mrež, je pomemben tudi izbor števila iteracij (angl. epoch) in velikost učnega paketa (angl. batch), saj lahko pride do prevelikega prileganja, kar upočasni učenje ciljne mreže.



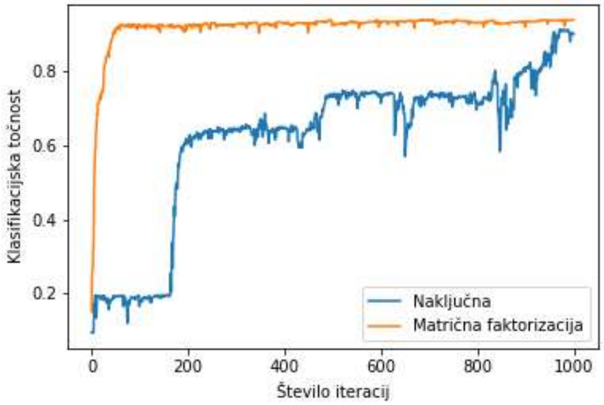
Slika 1. Izgradnja in učenje enonivojskih mrež ter njihovo združevanje v ciljno nevronsko mrežo.

Zadnji korak inicializacije je, da uteži teh enonivojskih mrež uporabimo za inicializacijo ciljne globoke nevronske mreže. Slika 1 prikazuje postopek, kjer enonivojske mreže združimo v ciljno mrežo, ki jo še dodatno učimo.

3. REZULTATI MNIST

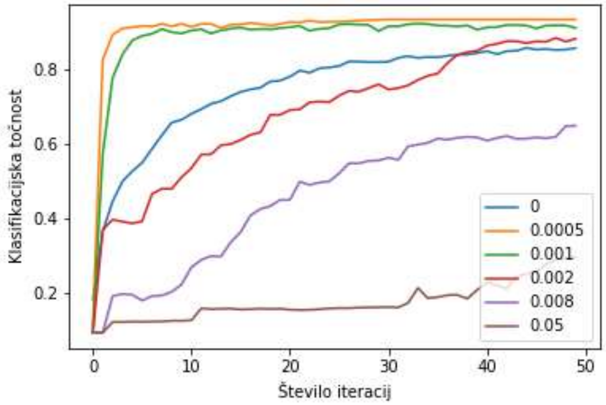
Za testiranje inicializacije za klasifikacijske probleme je bila uporabljena podatkovna množica MNIST, ki vsebuje slike ročno napisanih številk. Slika 2 prikazuje hitrost učenja ciljne nevronske mreže inicializirane s predlagano inicializacijo. Razvidno je, da za globoko nevronsko mrežo učenje poteka precej hitreje kot inicializacija z naključno inicializacijo Xavier.

Pri nevronskih mrežah z manj kot 5 skritimi nivoji, je ta inicializacija največkrat nepotrebna, saj med učenjem ne prihaja do tako velikih izgub gradienta, kot je to pri globokih nevronskih mrežah.



Slika 2. Primerjava klasifikacijske točnosti med predlagano inicializacijo in naključno inicializacijo Xavier na globoki nevronski mreži z arhitekturo nivojev (400, 300, 200, 100, 70, 50, 40, 30, 20, 15, 13, 10) in sigmoidno aktivacijsko funkcijo.

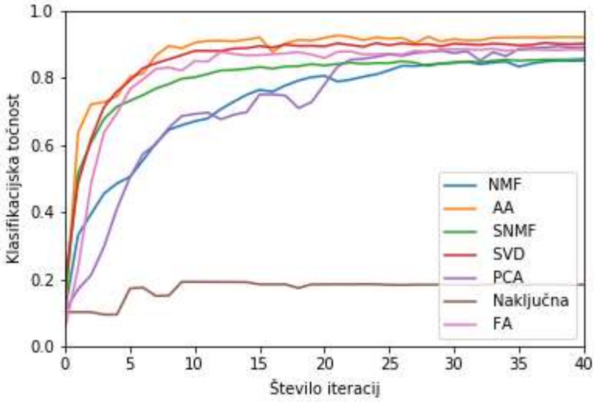
Ker lahko pri enonivojskih nevronskih mrežah pride do prevelikega prileganja podatkom, kar slabo vpliva na klasifikacijsko točnost ciljne mreže, lahko že pri učenju teh mrež dodamo regularizacijo, ki nekoliko prepreči preveliko prileganje ciljne mreže. Slika 3 prikazuje vpliv različnih stopenj regularizacije L2 na točnost ciljne mreže.



Slika 3. Vpliv različnih stopenj regularizacije L2 enonivojskih mrež na klasifikacijsko točnost med učenjem ciljne nevronske mreže.

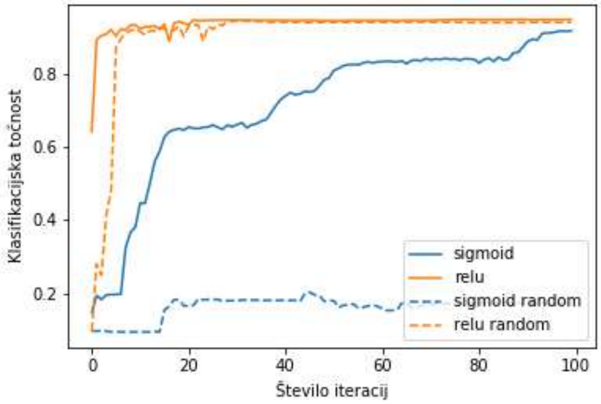
Iz stopenj regularizacije je razvidno, da lahko ob pravilni stopnji regularizacije enonivojskih mrež pospešimo hitrost učenja ciljne nevronske mreže. Če je regularizacija enonivojskih mrež prevelika, se te ne naučijo uporabne preslikave, kar zmanjša hitrost učenja ciljne mreže.

Pri izračunu zgoščene matrike podatkov lahko uporabimo več različnih metod za matrično faktorizacijo. Metode faktorizacije se med seboj precej razlikujejo. Slika 4 prikazuje vpliv različnih metod matrične faktorizacije na klasifikacijsko točnost ciljne nevronske mreže med učenjem. Uporabljeni so bili tipi NMF (nenegativne matrična faktorizacija), AA (analiza arhetipov) [1], SNMF (semi-nenegativna matrična faktorizacija), SVD (singularni razcep), PCA (analiza glavnih komponent) in FA (faktorska analiza). Za primerjavo je zraven dodana še inicializacija Xavier, ki se v tem primeru odreže precej slabše kot druge inicializacije. Pri napovedovanju se najbolje odreže algoritem AA, ki najhitreje doseže najboljšo klasifikacijsko točnost. Najslabša sta običajno algoritma PCA in NMF.



Slika 4. Primerjava vpliva različnih tipov matrične faktorizacije na hitrost učenja ciljne nevronske mreže.

Ena izmed najboljših postopkov za pohitritev učenja in preprečevanje nasičenosti aktivacijskih funkcij je uporaba funkcij, kot so ReLU ali katerokoli izmed mnogih variant te funkcije. Slika 5 primerja hitrost učenja med aktivacijsko funkcijo ReLU in sigmoidno aktivacijsko funkcijo.



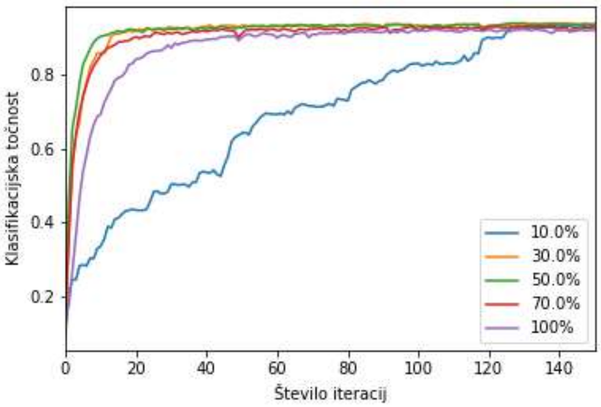
Slika 5. Primerjava spreminjanje klasifikacijske točnosti pri uporabi aktivacijske funkcije ReLU in sigmoidne funkcije.

Izbor aktivacijske funkcije zelo vpliva na hitrost učenja, kar je povsem pričakovano. Pri uporabi sigmoidne funkcije se predlagana inicializacija obnese precej dobro, saj se že po 100 iteracijah nauči precej dobre inicializacije, med tem ko se pri inicializaciji Xavier klasifikacijska točnost ne povzpne nad 20%.

Pri uporabi funkcije ReLU z inicializacijo dosežemo hitrejše učenje, vendar kar je najbolj zanimivo, že pred začetkom učenja ciljne mreže dosega ta klasifikacijo nad 65%, med tem ko je točnost pri naključni inicializaciji pred začetkom učenja ciljne mreže 10%.

Inicializacija s pomočjo matrične faktorizacije je precej počasen postopek, saj večkrat faktoriziramo celotno matriko ter nato še učimo enonivojske mreže. Slika 6 prikazuje inicializacijo, kjer smo za faktorizacijo in učenje enonivojskih mrež uporabili le del učne množice.

Iz grafa je vidno, da lahko pri uporabi le dela učne množice za inicializacijo dobimo boljše rezultate kot pa pri uporabi celotne množice, pri tem pa precej pridobimo še na hitrosti pri matrični faktorizaciji in hitrosti učenja enonivojskih mrež.

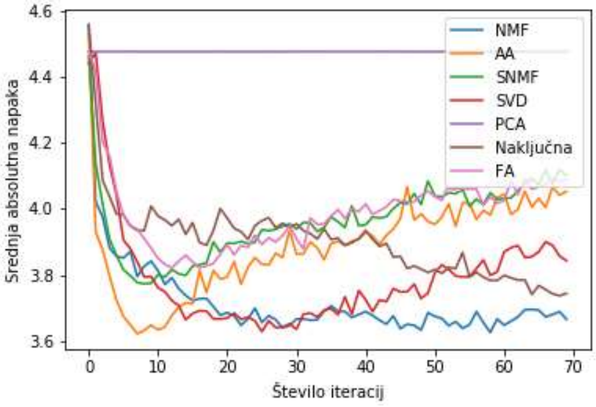


Slika 6. Primerjava klasifikacijske točnosti med učenjem ciljne mreže v odvisnosti od odstotka uporabljenih podatkov pri inicializaciji.

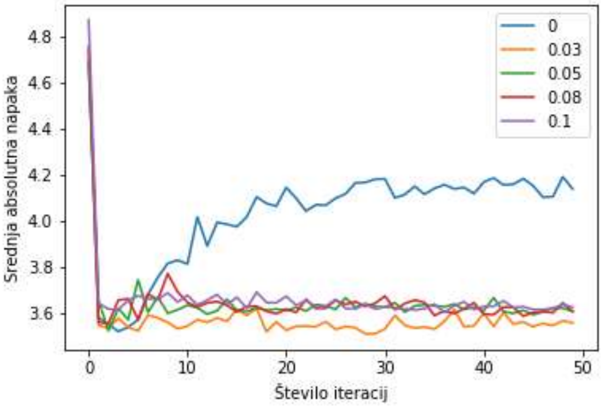
4. REZULTATI JESTER JOKES

Inicializacija je bila testirana tudi za regresijski problem napovedovanja ocen šal, ki bi jih uporabniki dali šalam na spletni strani Jester.

Slika 7 prikazuje hitrost učenja pri uporabi različnih tipov matrične faktorizacije. Najboljša je bila ponovno faktorizacija AA, najslabša pa PCA, ki se sploh ni učila.



Slika 7. Primerjava vpliva različnih tipov matrične faktorizacije na hitrost učenja ciljne nevronske mreže.



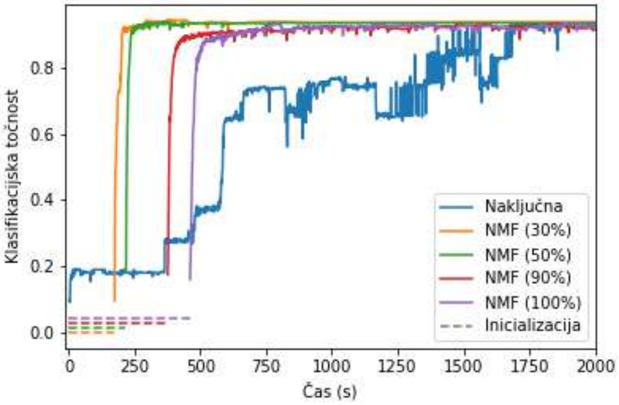
Slika 8. Napaka MAE ciljne mreže pri uporabi različnih stopenj regularizacije L1 enonivojskih mrež.

Največja prednost predlagane inicializacije se je izkazala pri regularizaciji šal, saj je ciljna nevronska mreža pri šalah dosegla optimalno povprečno absolutno napako že po nekaj iteracijah, nato pa je prišlo do prevelikega prileganja. Slika 8 prikazuje MAE ciljne nevronske mreže pri različnih stopnjah regularizacije L1 enonivojskih nevronskih mrež.

Iz napake MAE vidimo, da lahko že s pravilno nastavljenimi utežmi, ki jih dobimo z regularizacijo uteži enonivojskih mrež dosežemo podoben efekt, kot če bi regularizacijo uporabili med učenjem ciljne nevronske mreže.

5. ČAS, POTREBEN ZA INICIALIZACIJO

Pri predlagani inicializaciji je potrebno najprej izvesti večje število matričnih faktorizacij, ki odvisno od časovne kompleksnosti potrebujejo precej časa. Ko končamo s faktorizacijo je potrebno še učenje enonivojskih mrež. V določenih primerih je za inicializacijo potrebno celo več časa kot pa za učenje ciljne nevronske mreže. Slika 9 primerja skupen čas, potreben za inicializacijo in učenje ciljne nevronske mreže, pri različnih odstotkih uporabljenih učnih podatkov za inicializacijo v odvisnosti od klasifikacijske točnosti ciljne mreže. Za primerjavo so bili uporabljeni podatki MNIST. Črtkana črta prikazuje čas, ki je potreben za inicializacijo in je odvisen od odstotka učne množice, uporabljene za inicializacijo. Pri uporabi le 30% podatkov se inicializacija izvede že po 150 sekundah, medtem ko inicializacija pri uporabi celotne množice potrebuje skoraj 500 sekund. Naključna inicializacija Xavier je na začetku boljša od predlagane inicializacije, saj že hitro po začetku učenja dosega 20% točnost, vendar je učenje precej počasno, zato jo druge nevronske mreže, ko so enkrat inicializirane hitro prehitijo po kriteriju klasifikacijske točnosti. Za merjenje časa je učenje nevronskih mrež potekalo z grafično kartico GTX 960M. Za faktorizacijo in učenje nevronskih mrež pa so bile uporabljene knjižnice sklearn, PyMF in Keras [2].



Slika 9. Primerjava časa, ki ga potrebuje predlagana inicializacija za izračun uteži v odvisnosti od klasifikacijske točnosti med učenjem.

Matriki W in H sta pred začetkom faktorizacije inicializirani naključno. Prav tako so naključno inicializirane uteži enonivojskih mrež. Zaradi naključne inicializacije so lahko rezultati med posameznimi testi nekoliko različni, zato so vsi grafi dobljeni kot povprečje treh poganjanj algoritma. Isto velja za graf časov, ki so sestavljeni kot povprečje treh testov.

6. ZAKLJUČEK

Predstavljena in testirana je inicializacija, kjer začetne vrednosti uteži izračunamo s pomočjo matrične faktorizacije. Metoda se dobro obnese pri podatkih, ki imajo precej veliko število atributov in za napovedovanje izhodnih vrednosti potrebujejo globoke nevronske mreže, saj se mreža že v začetku nauči nekatere abstraktne koncepte, ki obstajajo v podatkih.

7. PODOBNE INICIALIZACIJE

Skozi razvoj nevronskih mrež so te postajale vsakič bolj globoke, da bi se lahko naučile čim bolj zapletenih povezav med podatki. Pri globljih mrežah se pri naključni inicializaciji pojavi težava med učenjem, saj hitro prihaja do izgube gradientov. Ena izmed metod, ki pospeši učenje, je naključna inicializacija Xavier, ki uteži skalira, da se med učenjem aktivacijske funkcije ne nasičijo tako hitro. Druge možnosti za preprečevanje nasičenih funkcij so še Batch Normalization [3] in Self-Normalizing Neural Networks [8], kjer izhode nevronov normaliziramo med učenjem.

Predlagana inicializacija je najbolj podobna inicializacijama uteži, imenovanima Deep Autoencoder [4] ali Deep Belief Network [5], kjer je cilj, da mreži z nenadzorovanim učenjem nastavimo čim boljše uteži, ki nam bodo pomagale optimizaciji uteži. Prednost te inicializacije je ta, da lahko faktorizacijo in učenje enonivojskih mrež izvajamo paralelno, kar pa ni možno pri drugih dveh inicializacijah.

8. NADALJNE DELO

- Paralelizacija matrične faktorizacije in učenja enonivojskih mrež.
- Izračun matrike W_{k+1} s faktorizacijo matrike W_k namesto matrike X.
- Pretvorba inicializacije, ki bi delovala s konvolucijskimi nevronskimi mrežami.

9. VIRI

[1] A. Cutler and L. Breiman, "Archetypal analysis," *Technometrics,* vol. 36, pp. 338-347, 1994.

[2] F. Chollet and others, *Keras,* GitHub, 2015.

[3] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning,* 2015.

[4] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of Machine Learning Research,* vol. 11, pp. 3371-3408, 2010.

[5] G. E. Hinton, S. Osindero and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation,* vol. 18, pp. 1527-1554, 2006.

[6] A. Graves, A. r. Mohamed and G. Hinton, "Speech recognition with deep recurrent neural networks," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing,* 2013.

[7] I. Sutskever, O. Vinyals and Q. V. Le, "Sequence to Sequence Learning with Neural Networks," in *Advances in Neural Information Processing Systems 27,* Curran Associates, Inc., 2014, pp. 3104-3112.

[8] G. Klambauer, T. Unterthiner, A. Mayr and S. Hochreiter, "Self-Normalizing Neural Networks," *CoRR,* vol. abs/1706.02515, 2017.

Optimiranje časa in porabe goriva v modelih človeške vožnje

Erik Dovgan
Fakulteta za računalništvo in informatiko
Univerza v Ljubljani
Večna pot 113, 1000 Ljubljana
Odsek za inteligentne sisteme
Institut „Jožef Stefan“
Jamova cesta 39, 1000 Ljubljana
erik.dovgan@fri.uni-lj.si

Jaka Sodnik
Fakulteta za elektrotehniko
Univerza v Ljubljani
Tržaška cesta 25, 1000 Ljubljana
NERVteh, raziskave in razvoj, d.o.o.
Kidričeva ulica 118, 1236 Trzin

Ivan Bratko
Fakulteta za računalništvo in informatiko
Univerza v Ljubljani
Večna pot 113, 1000 Ljubljana

Bogdan Filipič
Odsek za inteligentne sisteme
Institut „Jožef Stefan“
Jamova cesta 39, 1000 Ljubljana

POVZETEK

Ko vozniki vozijo po cesti, optimirajo več kriterijev, npr. čas vožnje in porabo goriva. Toda teh kriterijev se navadno ne upošteva pri gradnji modelov človeške vožnje. Za namen sočasne optimizacije tako človeških vidikov vožnje kot kriterijev vožnje smo razvili Večkriterijski optimizacijski algoritem za iskanje strategij vožnje podobnih človeškim (ang. Multiobjective Optimization algorithm for discovering Human-like Driving Strategies, MOHDS). Algoritem vključuje modele človeške vožnje in optimira tri kriterije: čas vožnje, porabo goriva in podobnost s človeškimi vožnjami. MOHDS smo ovrednotili na treh cestah, ki so vključevale ovinke, naklone, druga vozila in avtocesto. Dobljene strategije vožnje smo primerjali s človeškimi strategijami vožnje. Rezultati kažejo, da MOHDS najde strategije vožnje, ki so z vidika kriterijev primerljive s človeškimi strategijami vožnje v večini obravnavanih scenarijev vožnje.

Ključne besede

večkriterijska optimizacija, človeške strategije vožnje, čas vožnje, poraba goriva

1. UVOD

Avtonomna vožnja vozil je zelo aktivno raziskovalno področje, na katerem deluje mnogo znanih podjetij, kot sta Google [11] in Toyota [9]. Veliko sistemov za pomoč voznikom, kot je npr. sistem za ohranjanje voznega pasu, je že vgrajenih v sodobna vozila. Poleg tega popolnoma avtonomna vozila že vozijo po javnih cestah [7].

Sistemi za avtonomno vožnjo se osredotočajo na zaznavanje okolice, kar vključuje druga vozila, pešce, obliko ceste, razne ovire na cesti itd. Toda dobljena vožnja lahko ne zadošča ostalim kriterijem vožnje, kot so čas vožnje, poraba goriva in posledično onesnaževanje okolja, udobje, podobnost s človeško vožnjo itd. Ti kriteriji vplivajo na sprejemljivost avtonomne vožnje s strani potnikov. Na primer, potniki ne želijo, da bi bila avtonomna vožnja preveč nenavadna, zelo različna od njihove vožnje ali pa slabša od človeške vožnje [12].

Ta prispevek opisuje dvonivojski Večkriterijski optimizacijski algoritem za iskanje strategij vožnje podobnih človeškim (MOHDS), ki vključuje modele za oponašanje človeške vožnje ter optimira čas vožnje, porabo goriva in človeškost vožnje. Algoritem na spodnjem nivoju vključuje množico matematičnih modelov, ki oponašajo človeško vožnjo. Algoritem na zgornjem nivoju pa je večkriterijski optimizacijski algoritem, razvit na podlagi algoritmov Non-dominated Sorting Genetic Algorithm II (NSGA-II) [1] in Differential Evolution for Multiobjective Optimization (DEMO) [10], ki išče najboljše vrednosti parametrov za algoritem na spodnjem nivoju. Prispevek predstavi tudi okolje za simulacijo vožnje, s katerim smo vrednotili algoritem MOHDS.

Prispevek je nadalje organiziran kot sledi. Razdelek 2 opisuje sorodno delo na področju avtonomne vožnje. Okolje za simulacijo vožnje je predstavljeno v razdelku 3. Razdelek 4 opisuje algoritem MOHDS. Poskusi in rezultati so navedeni v razdelku 5. Prispevek zaključimo s povzetkom opravljenega dela in napovedjo nadaljnjega dela v razdelku 6.

2. SORODNO DELO

Človeške strategije vožnje lahko posnemamo z uporabo modelov človeške vožnje, pri čemer se obstoječi modeli osredotočajo na specifične aktivnosti vožnje, kot so sledenje vozilom, prosta vožnja, prehitevanje, sprememba pasu itd. Modeli za sledenje vozilom opisujejo aktivnost sledenja predhodnim vozilom na istem pasu. Ti modeli predpisujejo, da sledeče vozilo pospešuje oziroma zavira kot odziv na podatke iz okolice, pri čemer se modeli razlikujejo v upoštevanju teh podatkov. Na splošno lahko podatki vključujejo hitrost in pospešek vozila, relativno hitrost glede na predhodno vozilo, razdaljo do predhodnega vozila itd. [8, 14]. Modeli za prehitevanje na regionalni cesti in spremembo pasu na avtocesti opisujejo odločitveni proces za ustrezne aktivnosti vožnje. Spremembo pasu običajno modelirajo z modelom želje po spremembi pasu, modelom za sprejem vrzeli in modelom za izbiro vrzeli [13]. Prehitevanje modelirajo z modelom želje po prehitevanju in modelom za sprejem vrzeli [3]. Ti modeli so organizirani zaporedno, tj. za spremembo pasu najprej preverijo željo po spremembi pasu, ob zadostni želji preverijo, ali je vrzel zadostna, ter če je zadostna vsaj ena vrzel, izberejo najustreznejšo vrzel. Podoben postopek je tudi pri prehitevanju, le da ne vključuje izbire vrzeli.

Modeli človeške vožnje posnemajo človeško obnašanje, pri čemer pa zanemarijo ostale kriterije, ki so tudi pomembni med vožnjo, kot so čas vožnje in poraba goriva. Razvitih je bilo več pristopov za optimiranje teh kriterijev, ki večinoma vključijo vse kriterije v eno kriterijsko funkcijo ali pa optimirajo samo porabo goriva, čas vožnje pa vključijo kot omejitev. Hellstrom in sod. [4] so razvili metodo dinamičnega programiranja, ki optimira uteženo vsoto kriterijev. Razvitih je bilo tudi več analitičnih metod za optimiranje utežene vsote kriterijev [6] oziroma le porabe goriva [5].

Obstoječe metode za iskanje strategij vožnje se osredotočajo bodisi na strategije vožnje podobne človeškim, bodisi na optimizacijo časa vožnje, porabe goriva in/ali ostalih kriterijev. Ta prispevek predstavlja algoritem, ki rešuje oba problema hkrati: modelira človeško vožnjo z modeli človeške vožnje ter uglašuje parametre teh modelov, pri čemer optimira čas vožnje, porabo goriva in podobnost s človeškimi vožnjami.

3. SIMULACIJA VOŽNJE

Simulacijsko okolje je namenjeno vrednotenju strategij vožnje in omogoča simulacijo vožnje po regionalni cesti in dvopasovni avtocesti. Cesta je razdeljena na odseke, pri čemer za vsak odsek določimo dolžino, omejitev hitrosti, polmer ovinka, naklon, smer vožnje pasov in možnost prehitevanja. Na cesti so lahko druga vozila, ki vozijo v smeri pasu ter ne prehitevajo. Tem vozilom določimo hitrost in razdaljo do predhodnih vozil.

Vozilo vodimo z ukrepi vodenja, ki vključujejo pospešek, kot vozila glede na smer ceste in prestavo. Prestava se spreminja glede na spodnjo in zgornjo mejo hitrosti motorja. Pri vožnji vozila upoštevamo tudi fizikalne omejitve vozila, s katerimi določimo največji pospešek, ki ga lahko vozilo doseže.

Simulacijo izvajamo v več korakih, dokler vožnja po celotni cesti ni končana. V vsakem koraku preverimo veljavnost vožnje, tj. vozilo se ne sme ustaviti, se ne sme zaleteti in ne sme kršiti omejitve hitrosti. Za celotno vožnjo izračunamo naslednje kriterije: čas vožnje, porabo goriva in podobnost s človeško vožnjo. Porabo goriva izračunamo na podlagi diagrama specifične porabe goriva. Podobnost s človeško vožnjo izračunamo na podlagi vnaprej pridobljenih podatkov o vožnjah voznikov. Ker pa želimo minimizirati vse kriterije, namesto podobnosti izračunamo različnost glede na človeško vožnjo. Za vsakega voznika izračunamo različnost s srednjim kvadratnim odklonom (ang. Root Mean Square Error, RMSE) in vrnemo RMSE voznika z najmanj različno vožnjo. Za ta izračun upoštevamo dva atributa: hitrost in odmik od sredine desnega voznega pasu. Pri tem izračunamo RMSE vsakega atributa in upoštevamo povprečje.

4. ALGORITEM ZA ISKANJE STRATEGIJ VOŽNJE PODOBNIH ČLOVEŠKIM

Ta razdelek opisuje Večkriterijski optimizacijski algoritem za iskanje strategij vožnje podobnih človeškim (MOHDS), ki sočasno oponaša človeško vožnjo in optimira čas vožnje, porabo goriva in različnost od človeške vožnje. MOHDS sestoji iz dveh nivojev. Na spodnjem nivoju je implementiranih več modelov človeške vožnje, ki vodijo vozilo v različnih aktivnostih vožnje. Vrednosti parametrov modelov na spodnjem nivoju iščemo z algoritmom na zgornjem nivoju, tj. večkriterijskim optimizacijskim algoritmom, ki minimizira čas vožnje, porabo goriva in različnost od človeških strategij vožnje. Začetna verzija algoritma je bila predstavljena v [2]. Ta verzija je sedaj nadgrajena z modeli za vožnjo po klancih in ovinkih, modeli za spreminjanje pasu ter s primerjavo s človeškimi strategijami vožnje oziroma optimizacijo glede na različnost od človeških strategij vožnje.

4.1 Algoritem na spodnjem nivoju

Algoritem na spodnjem nivoju vključuje množico matematičnih modelov, ki oponašajo človeško vožnjo in upravljajo vozilo pri naslednjih aktivnostih vožnje: (a) prosta vožnja, (b) sledenje vozilom, (c) zaviranje v sili, (d) prehitevanje in (e) sprememba pasu. Modeli za sledenje vozilom, prosto vožnjo in zaviranje v sili določajo pospešek vozila, medtem ko modeli za prehitevanje in spremembo pasu odločajo, kdaj vozilo spremeni vozni pas.

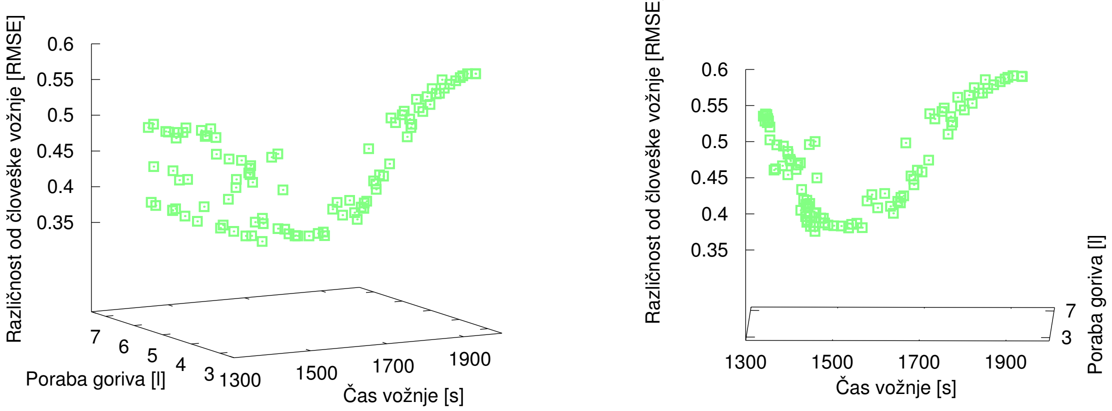
Model za sledenje vozilom temelji na modelu Gazis-Herman-Rothery (GHR) [8], ki določa pospešek vozila glede na hitrost vozila, razdaljo do predhodnega vozila in razliko hitrosti med vozilom in predhodnim vozilom. Ko je vozilo daleč od predhodnega vozila, uporabimo namesto modela za sledenje model za prosto vožnjo, ki vodi vozilo s konstantnim pospeškom, dokler ni dosežena ciljna hitrost. Ko je vozilo preblizu predhodnemu vozilu, uporabimo model za zaviranje v sili. Poleg zgornjih modelov uporabimo za določanje pospeška še naslednje omejitve. Pospešek se lahko zmanjša glede na omejitve vozila, kot je opisano v razdelku 3. Za vsak vozni pas ima vozilo določeno ciljno hitrost. Poleg tega se ciljna hitrost voznega pasu zmanjša, če je cestni odsek ovinek ali klanec, pri čemer je stopnja zmanjšanja ciljne hitrosti odvisna od ostrine ovinka in naklona odseka.

Prehitevanje na regionalni cesti je določeno z modeloma želje po prehitevanju in za sprejem vrzeli [3]. Model želje po spremembi pasu na podlagi ciljne hitrosti, razdalje do predhodnega vozila in hitrosti predhodnega vozila določi, kdaj vozilo želi prehiteti. Nato preverimo sprejemljivost prednje vrzeli na drugem pasu na podlagi hitrosti vozila, hitrosti predhodnega vozila na istem pasu in hitrosti predhodnega vozila na drugem pasu.

Sprememba pasu na avtocesti je določena z modelom želje po spremembi pasu in modelom za sprejem vrzeli [13]. Podobno kot pri prehitevanju tudi ta model želje po spremembi pasu na podlagi hitrosti vozila, razlike hitrosti glede na predhodno vozilo in razlike hitrosti glede na predhodno vozilo na drugem pasu določi, kdaj vozilo želi spremeniti vozni pas. Nato preverimo sprejemljivost prednje in zadnje vrzeli na drugem pasu glede na razlike hitrosti glede na predhodno vozilo in vozilo zadaj na drugem voznem pasu.

4.2 Algoritem na zgornjem nivoju

Algoritem na zgornjem nivoju je večkriterijski optimizacijski algoritem, razvit na podlagi algoritmov NSGA-II [1] in DEMO [10]. Algoritem išče najboljše vrednosti parametrov



Slika 1: Strategije vožnje v kriterijskem prostoru, najdene z algoritmom MOHDS: dva različna pogleda.

modelov na spodnjem nivoju, pri čemer minimizira čas vožnje, porabo goriva in različnost od človeških voženj. Za iskanje uporablja evolucijski pristop, ki množico rešitev izboljšuje skozi več generacij. Za ohranjanje konstantne velikosti populacije uporablja nedominirano razvrščanje in metriko nakopičenosti iz algoritma NSGA-II [1]. Rezultat optimizacije je množica nedominiranih rešitev. Za več podrobnosti glej [2].

5. POSKUSI IN REZULTATI

MOHDS smo vrednotili na treh cestah in rezultate primerjali s človeškimi vožnjami. Naslednji razdelki opisujejo testne scenarije in dobljene rezultate.

5.1 Opis poskusov

Vrednotenje algoritma MOHDS smo izvedli na naslednjih cestah: (a) Regionalna cesta dolžine 11.450 m, na kateri ni drugih vozil. Cesta vključuje različne omejitve hitrosti, tri stopnje ovinkov (od blagega do zelo ostrega) ter tri stopnje klancev navzgor in navzdol (od blagega do zelo strmega); (b) Dvopasovna regionalna cesta dolžine 9.000 m, po kateri vozijo tudi druga vozila. Cesta je brez ovinkov in klancev ter ima konstantno omejitev hitrosti. Vozila na desnem pasu spreminjajo hitrost, medtem ko imajo vozila na levem pasu, ki vozijo v nasprotno smer, konstantno hitrost. Razdalja med vozili na levem pasu je na začetku ceste krajša, nato pa daljša. Cesta se začne s polno sredinsko črto, na približno polovici ceste pa sredinska črta postane prekinjena; (c) Dvopasovna avtocesta dolžine 10.000 m, po kateri vozijo tudi druga vozila. Cesta je brez ovinkov in klancev ter ima konstantno omejitev hitrosti. Na celotni cesti je prekinjena sredinska črta. Vozila na desnem pasu spreminjajo hitrost, medtem ko imajo vozila na levem pasu, ki vozijo v isto smer, konstantno hitrost. Razdalja med vozili na levem pasu je na začetku ceste krajša, nato pa daljša.

Za vsako cesto smo pridobili podatke o človeških vožnjah 30 voznikov, pri čemer je vsak voznik prevozil vsako cesto dvakrat. Različnost od človeške vožnje smo nato izračunali za vsakega voznika posebej kot povprečje vseh voznikovih voženj, kot je opisano v razdelku 3. Različnost po hitrosti smo ocenjevali na prvi cesti in prvi polovici druge ceste, medtem ko smo različnost po odmiku od sredine desnega voznega pasu preverjali na drugi polovici druge ceste ter na tretji cesti. Vsako strategijo vožnje smo vrednotili na vseh treh cestah in pri iskanju strategij optimirali skupni čas vožnje, skupno porabo goriva in skupno različnost od človeške vožnje. Večkriterijsko optimizacijo smo izvajali skozi 200 generacij, pri čemer je bila velikost populacije 100 rešitev. Za ostale parametre algoritma MOHDS glej [2].

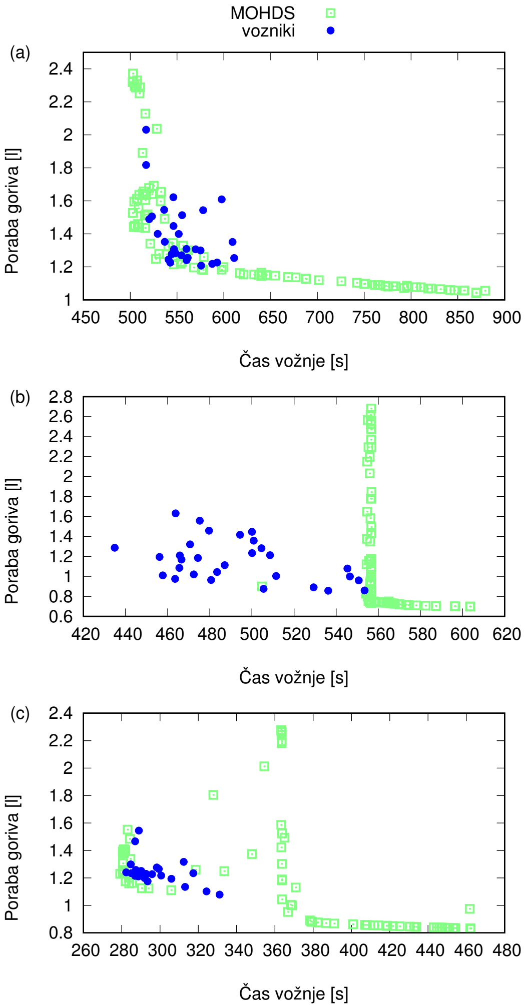
5.2 Rezultati

Strategije vožnje, pridobljene z algoritmom MOHDS, so prikazane na sliki 1. Rezultati kažejo, da MOHDS najde strategije vožnje z različnimi kompromisi med kriteriji, tj. časom vožnje, porabo goriva in različnostjo od človeške vožnje. Poleg tega se strategije vožnje, ki imajo bodisi najkrajši čas vožnje bodisi najnižjo porabo goriva, najbolj razlikujejo od človeške vožnje.

Slika 2 prikazuje primerjavo človeških strategij in strategij algoritma MOHDS glede na čas vožnje in porabo goriva. Kljub temu, da smo optimirali skupni čas vožnje, skupno porabo goriva in skupno različnost od človeške vožnje za vse tri ceste, so na tej sliki prikazane delne vrednosti kriterijev za vsako cesto posebej. Posledično so lahko določene strategije slabe na posamezni cesti, a so vseeno nedominirane glede na skupne vrednosti kriterijev za vse tri ceste. Ti rezultati kažejo, da MOHDS najde strategije vožnje, ki so primerljive s človeškimi strategijami vožnje na prvi in tretji cesti. Poleg tega kažejo, da MOHDS uspešneje optimira porabo goriva, tj. najde strategije vožnje z nižjo porabo goriva v primerjavi z vozniki. Dodatna analiza strategij vožnje algoritma MOHDS na drugi cesti je pokazala, da strategije prehitevajo bolj redko kot ljudje, kar onemogoča dodatno krajšanje časa vožnje.

6. ZAKLJUČEK

V prispevku smo predstavili dvonivojski Večkriterijski optimizacijski algoritem za iskanje strategij vožnje podobnih človeškim (MOHDS), ki sočasno optimira čas vožnje, porabo goriva in človeškost vožnje. Algoritem smo vrednotili z



Slika 2: Primerjava strategij vožnje algoritma MOHDS in voznikov glede na čas vožnje in porabo goriva na (a) prvi, (b) drugi in (c) tretji cesti.

vožnjo po treh cestah, tj. prazni regionalni cesti, regionalni cesti z drugimi vozili in avtocesti z drugimi vozili. Dobljene strategije vožnje smo primerjali z vožnjami voznikov. Rezultati kažejo, da MOHDS najde strategije vožnje z različnimi kompromisi med kriteriji. Poleg tega MOHDS najde podobne strategije vožnje kot vozniki na prazni regionalni cesti in na avtocesti, medtem ko na regionalni cesti z ostalimi vozili najde počasnejše strategije vožnje. V bodoče bomo dodatno analizirali in izboljševali algoritem MOHDS predvsem za vodenje vozil po regionalni cesti, kjer vozijo tudi druga vozila in je prehitevanje dovoljeno.

7. ZAHVALA

Projekt („Večkriterijsko iskanje strategij vožnje za avtonomna vozila“, št. Z2-7581) je sofinancirala Javna agencija za raziskovalno dejavnost Republike Slovenije iz državnega proračuna. Delo predstavljeno v tem prispevku je delno financiralo tudi podjetje NERVteh, raziskave in razvoj, d.o.o.

8. LITERATURA

[1] K. Deb, A. Pratap, S. Agrawal in T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

[2] E. Dovgan, J. Sodnik, I. Bratko in B. Filipič. Multiobjective discovery of human-like driving strategies. V *GECCO'17 Companion – Proceedings of the Genetic and Evolutionary Computation Conference GECCO 2017*, 8 strani, 2017.

[3] H. Farah in T. Toledo. Passing behavior on two-lane highways. *Transportation Research Part F*, 13(6):355–364, 2010.

[4] E. Hellstrom, J. Aslund in L. Nielsen. Design of an efficient algorithm for fuel-optimal look-ahead control. *Control Engineering Practice*, 18(11):1318–1327, 2010.

[5] P. G. Howlett, P. J. Pudney in X. Vu. Local energy minimization in optimal train control. *Automatica*, 45(11):2692–2698, 2009.

[6] M. Ivarsson, J. Aslund in L. Nielsen. Optimal speed on small gradients – consequences of a non-linear fuel map. V *Proceedings of the 17th World Congress of the International Federation of Automatic Control IFAC'08*, strani 3368–3373, 2008.

[7] E. Jaffe. The First Look at How Google's Self-Driving Car Handles City Streets, 2014. Dostopno na: `https://www.citylab.com/life/2014/04/first-look-how-googles-self-driving-car-handles-city-streets/8977/`.

[8] Y. Li in D. Sun. Microscopic car-following model for the traffic flow: the state of the art. *Journal of Control Theory and Applications*, 10(2):133–143, 2012.

[9] R. Read. Toyota will roll out autonomous cars by the 'mid-2010s', 2013. Dostopno na: `http://www.thecarconnection.com/news/1087636\_toyota-will-roll-out-autonomous-cars-by-the-mid-2010s`.

[10] T. Robič in B. Filipič. DEMO: Differential evolution for multiobjective optimization. V *Proceedings of the Third International Conference on Evolutionary Multi-Criterion Optimization – EMO 2005*, zvezek 3410, strani 520–533, 2005.

[11] R. J. Rosen. Google's self-driving cars: 300,000 miles logged, not a single accident under computer control, 2012. Dostopno na: `http://www.theatlantic.com/technology/archive/2012/08/googles-self-driving-cars-300-000-miles-logged-not-a-single-accident-under-computer-control/260926/`.

[12] B. Schoettle in M. Sivak. A Survey of Public Opinion about Autonomous and Self-Driving Vehicles in the U.S., the U.K., and Australia. Technical Report UMTRI-2014-21, Transportation Research Institute, The University of Michigan, 2014.

[13] T. Toledo, H. N. Koutsopoulos in M. E. Ben-Akiva. Modeling integrated lane-changing behavior. *Transportation Research Record*, 1857:30–38, 2003.

[14] T. H. Yang in C. W. Zu. Linear dynamic car-following model. V *Proceedings of the 5th World Congress on Intelligent Control and Automation*, zvezek 6, strani 5212–5216, 2004.

User-friendly Multi-objective Learning of Accurate and Comprehensible Hybrid-trees

Benjamin Novak
Ježef Stefan Institute,
Department of Intelligent Systems
Jamova cesta 39
1000, Ljubljana
benjaminovak@gmail.com

Rok Piltaver
Ježef Stefan Institute,
Department of Intelligent Systems
Jamova cesta 39
1000, Ljubljana
rok.piltaver@ijs.si

Matjaž Gams
Ježef Stefan Institute,
Department of Intelligent Systems
Jamova cesta 39
1000, Ljubljana
matjaz.gams@ijs.si

ABSTRACT

In data-mining applications, users must often choose between a comprehensible classifiers with low accuracy or a more accurate but incomprehensible classifier. A possible solution is to apply MOLHC algorithm, which finds the complete set of non-dominated hybrid-trees according to accuracy and comprehensibility. However, tools that would help the user select an appropriate hybrid-tree from the set are missing. Therefore, we present MOLHC implemented as an add-on for Orange data-mining suite. We implemented widgets for learning, evaluation and visualization of individual hybrid-trees and the set of non-dominated hybrid-trees. They comprise a user-friendly toolbox that enables users to efficiently execute the multi-objective data-mining process.

1. INTRODUCTION

There are two key criteria for selecting the classifier that will be deployed in a data-mining application: predictive performance and comprehensibility. The most appropriate measures of predictive performance (e.g. accuracy, area under the ROC curve, sensitivity, specificity) is used to select a subset of acceptable classifiers depending on the objectives of the application. However, acceptable predictive performance is often not enough. Comprehensibility is reported as decisive factor for classifier application in domains such as: medicine, credit scoring, churn prediction, and bioinformatics [2]. It is described as "the ability to understand the logic behind a prediction of the model" [4] or as "the ability to understand the output of induction algorithm" [3]. It is important because comprehensible classifiers enable explanation for classification of each instance, classifier validation, knowledge discovery and support classifier generalization improvement. Classification trees for example are often used in machine-learning applications because they are one of the most comprehensible classification models.

On the other hand, there are more complex classification models such as support vector machines, artificial neural networks and ensembles (e.g. random forest, boosting and stacking algorithms) that achieve higher accuracy then classification trees in many domains but are not comprehensible, hence they are referred to as black-box classifiers. Users are therefore faced with a difficult decision: they can either choose a comprehensible classification tree with relatively low accuracy or an incomprehensible classifier with relatively high accuracy. In many cases none of the two options is sufficient.

The difficulty of learning accurate and comprehensible classifiers arises from the fact that the two objectives must be considered as equally important and that they are conflicting: increasing one often decreases the other. The solution is multi-objective learning. It is based on multi-objective optimization and therefore returns a set of non-dominated classifiers (not a single classifier). This enables the user to improve the decision on the accuracy-comprehensibility trade-off by deferring it: instead of taking an arbitrary decision about the relative importance of the learning objective before the execution of the learning algorithm (and having to rerun it with different settings to obtain a classifier with a different trade-off) user can now take a well informed decision by simply comparing the subset of all non-dominated classifiers returned by the multi-objective algorithm.

This paper focuses on a recently developed example of such algorithm named Multi-objective learning of hybrid classifiers (MOLHC) [7]. It finds the entire Pareto set of hybrid-trees (according to accuracy and comprehensibility) by replacing sub-trees in the initial classification tree (given as an input) with black-box classifiers (also given as an input). The resulting hybrid-trees consist of regular leaves containing easily comprehensible rules and incomprehensible black-box leaves enabling improved accuracy. MOLHC was shown to produce classifiers with accuracy-comprehensibility trade-offs and offer insights into the data-mining task that are not available using traditional machine-learning algorithms as well as being fast and simple enough for applications in real-world problems [5].

After finding the entire Pareto front, the user has to select the best hybrid-tree, which was a cumbersome and time consuming task due to the lack of user-friendly tools to support it. Therefore, this paper presents a user-friendly implementation of the MOLHC algorithm and additional tools with a graphical user interface (GUI), which make learning, comparing and selecting the hybrid-trees and evaluating the results of learning efficient. The algorithm and the corresponding tools are implemented as an Orange add-on with three new Orange widgets shown in the Figure 1. The widgets are components in the visual programming environment called Orange Canvas [1]. Each widget offers a self contained data-mining functionality and a graphical user interface for setting its parameters and presenting its (main) results. A widget can be connected to other widgets by passing data from its output(s) to the input communication channel(s) of the other widgets, which enables visual programming of data-mining applications.

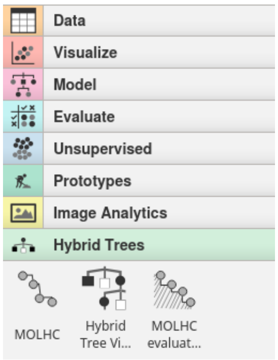


Figure 1: The list of developed Orange widgets.

The following sections describe the MOLHC related Orange widgets in the order in which they are typically applied during the data-mining process. Section 2 presents the MOLHC widget, which implements the MOLHC algorithm and visualizes the Pareto front of learned hybrid-trees. Section 3 presents the widget that visualizes the hybrid-tree chosen on the Pareto set and Section 4 presents the widget used for the evaluation of MOLHC results. Section 5 concludes the paper and discusses the ideas for future work.

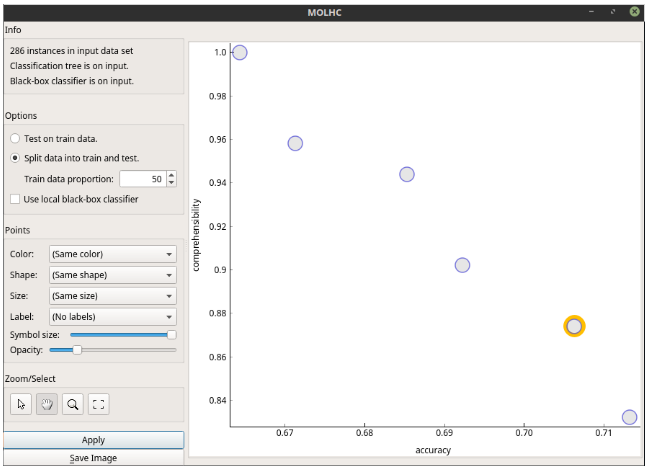
2. MOLHC WIDGET

The MOLHC widget implements Multi-objective learning of classifiers (MOLHC) algorithm. The widget finds and visualizes the entire Pareto-optimal set of hybrid-trees according to accuracy and comprehensibility as shown in Figure 2. Comprehensibility is defined as the share of instances that fall into the regular (not black-box) leaves of the hybrid-tree - this measure was designed based on the results of survey on the comprehensibility of classification trees [8].

Pareto-front viewer, which is a part of the widget's GUI (the right part in Figure 2), supports comparison and selection of an appropriate hybrid-tree from the set of learned hybrid-trees and enables extracting insights about the data-mining domain as discussed in our previous work [6].

The MOLHC widget is compatible with the standard Orange widgets so they can be used together as explained below and illustrated in Figure 3. Its inputs are:

- Data: data set for training and testing.



286 instances in input data set
Classification tree is on input.
Black-box classifier is on input.

○ Test on train data.
● Split data into train and test.
Train data proportion: 50
□ Use local black-box classifier

Color: (Same color)
Shape: (Same shape)
Size: (Same size)
Label: (No labels)
Symbol size:
Opacity:

Figure 2: MOLCH widget shows the Pareto front with 6 hybrid-trees (the chosen hybrid-tree is marked with an orange circle).

- Classification tree: the initial tree used by MOLCH.
- Black-box classifier: an (incomprehensible) classifier that has a higher accurate than the initial tree.

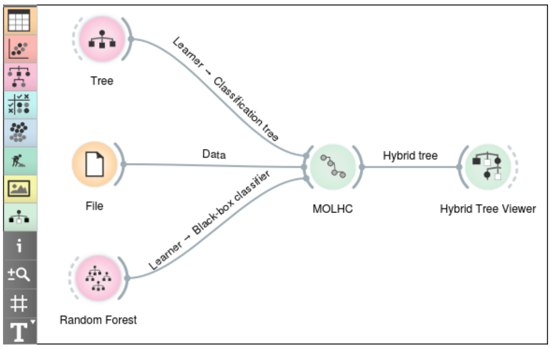


Figure 3: Connecting the MOLHC widget with the other widgets providing the required inputs (on the left) and accepting its output (on the right).

The output of the MOLHC widget is the hybrid-tree chosen by clicking on it in the Pareto-front viewer. The MOLHC widget offers the following options:

- Splitting options: splitting of the input data set into the training and testing set (used to estimate the accuracy and comprehensibility of the hybrid-trees on the Pareto-front) according to the set proportion or using all the data for both testing and training.
- Use local black-box classifier: use multiple black-box classifiers (each trained on the instances belonging to a specific leaf in the initial tree) instead of a single black-box classifier trained on all the training instances.
- Visualization options: zoom, selection and pan tool, change size or opacity of hybrid-tree symbols.

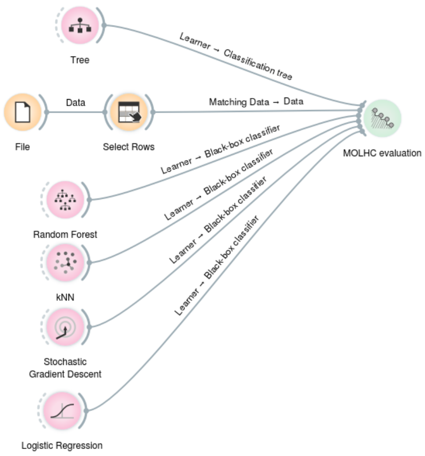

Tree
Learner → Classification tree
File
Data
Select Rows
Matching Data → Data
MOLHC evaluation
Learner → Black-box classifier
Random Forest
Learner → Black-box classifier
kNN
Learner → Black-box classifier
Stochastic Gradient Descent
Learner → Black-box classifier
Logistic Regression

Figure 4: Evaluation widget with inputs.

3. HYBRID-TREE VIEWER

The hybrid-tree viewer widget visualizes a hybrid-tree (an example is shown in Figure 5) received as an input from the MOLHC widget. The visualization of a hybrid-tree is used to validate it or extract knowledge from it. In addition, a pair of hybrid-tree viewer widgets positioned side by side can be used to compare a pair of hybrid-trees according to their structure; comparing them according to the accuracy and comprehensibility is done with the Pareto-front viewer in the MOLHC widget.

The parameters of the Hybrid-tree viewer define the display options, which are similar as the ones available in the standard Orange tree viewer. In addition, the user can choose whether to use the train or the test data set to compute the statistical data shown in the tree nodes.

The hybrid-tree viewer widget provides two outputs that depends on which node of the hybrid-tree the user selected by clicking on it:

- Selected data: the subset of the data set instances that belong to the selected tree node.
- Selected black-box model: the black-box classifier if a black-box leaf is chosen.

4. MOLHC EVALUATION

The MOLHC evaluation widget compares the Pareto set of hybrid-trees (i.e. the output of the MOLHC widget) with the set of baseline solutions consisting of the initial classification tree and the black-box classifier. It is used to evaluate the results of multi-objective learning and to select the best algorithm for learning the black-box classifier (used as the input to the MOLHC widget/algorithm). The evaluation is based on the hyper-volume measure [9], which is often used to compare results of multi-objective optimization algorithms.

The MOLHC evaluation widget requires at least three inputs (an example is shown in Figure 4). They are the same as te inputs of the MOLHC widget, except that multiple black-box classifiers can be provided for comparison.

The parameters of MOLHC evaluation widget are:

- The number of folds used for cross-validation.
- Train data proportion: the percentage of instances from the data set to be used as the training set, the rest of the instances are used as the testing set.
- Use local black-box classifier: use multiple black-box classifiers (one per leaf in the initial tree) instead of a single black-box classifier trained on all the training instances.

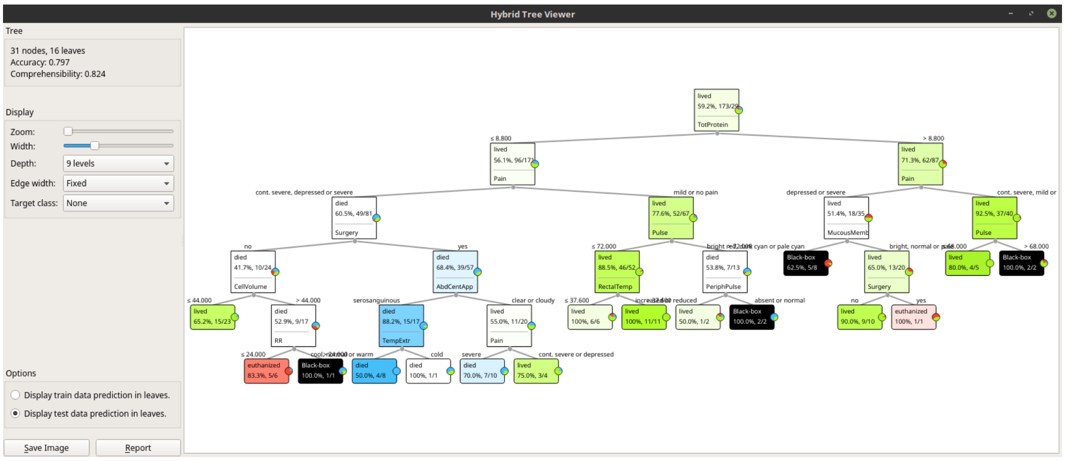
Figure 6 show an example of MOLHC evaluation. It is based on several measures for each provided black-box classifiers:

- Method: the name of the black-box classifier.
- BB accuracy: the accuracy of the black-box classifier.
- Tree accuracy: the accuracy of the initial tree.
- Baseline hypervolume: the hypervolume for the set of two baseline solutions (the initial tree and black-box classifier).
- MOLHC hypervolume: the hypervolume for the Pareto set of hybrid-trees learned by the MOLHC algorithm.
- Hypervolume difference: the difference between the MOLHC and the baseline hypervolume.

The results provided by the MOLHC evaluation widget are interpreted as follows. If the difference between the best BB accuracy and the tree accuracy is small, the user should consider using a regular classification tree instead of a black-box classifier or a hybrid-tree. In general a higher BB accuracy means that the corresponding black-box classifier has a higher potential to increase the accuracy of the initial classification tree. However, the actual results of the MOLHC algorithm depends on the accuracy of the black-box classifier in each leaf of the initial tree, therefore the best black-box classifier should be selected according to the MOLHC hypervolume. The overall success of the MOLHC algorithm is measured by the hypervolume difference - the higher the better. Finally, the user should select an appropriate hybrid-tree regardless of the hypervolume difference because it shows only the advantage of the MOLHC approach over the baseline algorithms, which depend on all the hybrid-trees in the Pareto set and is therefore not appropriate for the evaluation of a single hybrid-tree. To select a single hybrid-tree, the user should use Pareto front and hybrid-tree visualizer instead.

5. CONCLUSION

The paper presents an implementation of the MOLHC algorithm, which is a multi-objective learning algorithm that finds the complete set of non-dominated hybrid-trees according to accuracy and comprehensibility. The algorithm is most suitable for the data-mining applications where a



Hybrid Tree Viewer

Tree

31 nodes, 16 leaves
Accuracy: 0.797
Comprehensibility: 0.824

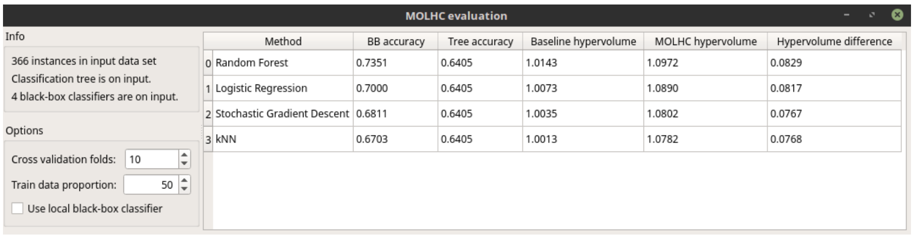
Display

Zoom:
Width:
Depth: 9 levels
Edge width: Fixed
Target class: None

Options

○ Display train data prediction in leaves.
● Display test data prediction in leaves.

Figure 5: Hybrid-tree viewer widget: the black-box leaves are marked with black.



MOLHC evaluation

Info

366 instances in input data set
Classification tree is on input.
4 black-box classifiers are on input.

Options

Cross validation folds: 10
Train data proportion: 50
☐ Use local black-box classifier

	Method	BB accuracy	Tree accuracy	Baseline hypervolume	MOLHC hypervolume	Hypervolume difference
0	Random Forest	0.7351	0.6405	1.0143	1.0972	0.0829
1	Logistic Regression	0.7000	0.6405	1.0073	1.0890	0.0817
2	Stochastic Gradient Descent	0.6811	0.6405	1.0035	1.0802	0.0767
3	kNN	0.6703	0.6405	1.0013	1.0782	0.0768

Figure 6: MOLHC evaluation widget with calculated results.

regular classification tree is not accurate enough and black-box classifier (with a higher accuracy) is not acceptable because it is not comprehensible. We implemented MOLHC as an add-on for Orange, which is a popular and user-friendly data-mining suite that offers visual programming and rich visualizations. We implemented three new Orange widgets for learning and visualizing the set of non-dominated hybrid-trees, visualizing individual hybrid-trees and for evaluation of the MOLHC results. They comprise a user-friendly toolbox that enables users to efficiently execute the multi-ob jective data-mining process. Nevertheless, we are considering several improvement that could make the developed widgets more user friendly. We plan to improve the documentation in order to make our work more accessibly to a wider user base and reduce the learning curve. Another improvement would be to color similar hybrid-trees on the Pareto front according to Hammilton distance which would help the user when comparing them. Finally, an improvement of the Hybrid-tree visualizer widget would add an option to replace the sub-trees that have only black-box leaves with a single black-box leaf.

6. REFERENCES

[1] J. Demšar, B. Zupan, G. Leban, and T. Curk. Orange: From experimental machine learning to interactive data mining. *Knowledge discovery in databases: PKDD 2004*, pages 537–539, 2004.

[2] A. A. Freitas. Comprehensible classification models: a position paper. *ACM SIGKDD explorations newsletter*, 15(1):1–10, 2014.

[3] R. Kohavi. Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In *KDD*, volume 96, pages 202–207, 1996.

[4] D. Martens, J. Vanthienen, W. Verbeke, and B. Baesens. Performance of classification models from a user perspective. *Decision Support Systems*, 51(4):782–793, 2011.

[5] R. Piltaver. *Constructing Comprehensible and Accurate Classifiers Using Data Mining Algorithms*. PhD thesis, Jožef Stefan international postgraduate school, 8 2016.

[6] R. Piltaver, M. Luštrek, and M. Gams. Multi-objective learning of accurate and comprehensible classifiers – a case study. In *Proceedings of 7th European Starting AI Researcher Symposium âĂŞ STAIRS 2014*, pages 220–229). IOS Press, 2014.

[7] R. Piltaver, M. Luštrek, J. Zupančič, S. Džeroski, and M. Gams. Multi-objective learning of hybrid classifiers. In *Proceedings of the Twenty-first European Conference on Artificial Intelligence*, pages 717–722. IOS Press, 2014.

[8] R. Piltaver, M. Luštrek, M. Gams, and S. Martinčić-Ipšić. What makes classification trees comprehensible? *Expert Systems with Applications*, 62(C):333–346, Nov. 2016.

[9] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. G. Da Fonseca. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on evolutionary computation*, 7(2):117–132, 2003.

Automatic Tennis Analysis with the use of Machine Learning and Multi-Objective Optimization

Miha Mlakar
Jožef Stefan Institute
Jamova cesta 39
1000 Ljubljana, Slovenia
miha.mlakar@ijs.si

Mitja Luštrek
Jožef Stefan Institute
Jamova cesta 39
1000 Ljubljana, Slovenia
mitja.lustrek@ijs.si

ABSTRACT

Wearable devices for monitoring players' movements are heavily used in many sports. However, the existing commercial and research sports wearables are either not tennis-specific, or are worn on the wrist or in the racquet and thus offer too limited information. We therefore added tennis-specific information to a leading commercial device. Our solution is two-fold. Firstly, we developed a model for classifying shot types into forehand, backhand and serve. Secondly, we designed an algorithm based on multi-objective optimization to distinguish active play from the time in-between points. By combining both parts with the general movement information already provided by the device, we get a comprehensive set of metrics that are used by professional tennis players and coaches to objectively measure a player's performance and enable in-depth tactical analysis.

Categories and Subject Descriptors

I.2.6. Artificial Intelligence: Training

General Terms

Algorithms, Measurement, Experimentation

Keywords

Tennis; Wearable analytics; Shot detection; Optimization;

1. INTRODUCTION

The use of wearable sensors in sport is growing fast and can already be considered essential for success in some disciplines. In tennis the analytics started with computer vision and sensors for measuring shots. However, both of these approaches have limitations for professional use. The sensors worn on the playing wrist or built into the tennis racquets deliver information about the shots [6] or enable the analysis and modeling of different shot techniques [7]. However, the problem with this information is the lack of context (under what circumstances and where on the court did a specific shot occur), so it is not sufficiently actionable, i.e. cannot be used for tactical preparations or to significantly improve players' game. Video analysis offers better information, and there has been a lot of research on this topic [1, 5]. However, cheap solutions offer low accuracy, while better solutions are extremely expensive because they require advanced cameras with complex software for calibration. Additionally, they are bound to a specific court, so the information is not available whenever needed by the player or coach.

Due to these limitations, devices worn on the torso, and equipped with accelerometers, gyroscopes and GPS receivers are emerging as the new approach. These devices are perfect for determining the effort, distance covered, sprints analysis and much more. Here, the leading provider in the world is Catapult Sports, whose S5 product is currently used by the best tennis player in the world Andy Murray. Nevertheless, the problem with S5 is that it offers no tennis-specific metrics. That is why in our research we add tennis-specific information to the metrics already available in the Catapult S5 system, to produce a comprehensive solution that enables professional players to make better tactical preparations and to improve their game.

Our algorithm consists of two parts. In the first part, we detect when a tennis shot occurs and which type of shot it is. In the second part, we focus on detecting when the players actually play points (active play) and when they are in-between points. This allows us to determine the actual net playing time and real distance covered and also adds context to shots which enable complex analysis like "Is the player playing weaker shots, if the point is longer than 15 seconds?" With this solution the players and their coaches get a continuous comprehensive view of the player's game, both the physical and the tactical part of it.

2. DATA AQUISITION

To obtain sensor data we used the commercially available S5 device from Catapult. The position of the device was high on the player's back attached to a tight shirt. The device contains a 3D accelerometer (frequency 100 Hz), 3D gyroscope (frequency 100 Hz), 3D magnetometer (frequency 100 Hz) and GPS sensor, returning latitude and longitude (frequency 10 Hz).

We recorded 5 different professional tennis players for 6 hours in total. Due to the 100 Hz frequency, we obtained 2,172,363 data records. In this time, we recorded 1,373 shots. Each shot was labeled as a serve, forehand or backhand. As for detecting active play, we also manually labeled the beginning and end of each sequence of active play. Because we were interested in creating an algorithm for detecting shot types and active plays in actual matches, all the data were recorded during matches and none during predefined situations of practice sessions.

3. SHOT DETECTION

For every data point obtained from our device, we extracted a number of features used by the shot detection algorithm. We used supervised machine learning to train a model to detect shots. With this model we classified every data point and evaluated the shot detection.

3.1 Feature Extraction

To define informative features for shot detection, we visualized and examined the traces for the accelerometer and gyroscope. Since we saw that every shot is associated with body rotation, our main source for feature extraction was the gyroscope - more specifically angular speeds on axes 1 (Roll) and 3 (Yaw) - and not the accelerometer. Figure 1 shows a typical trace of the gyroscope and accelerometer for a backhand shot.

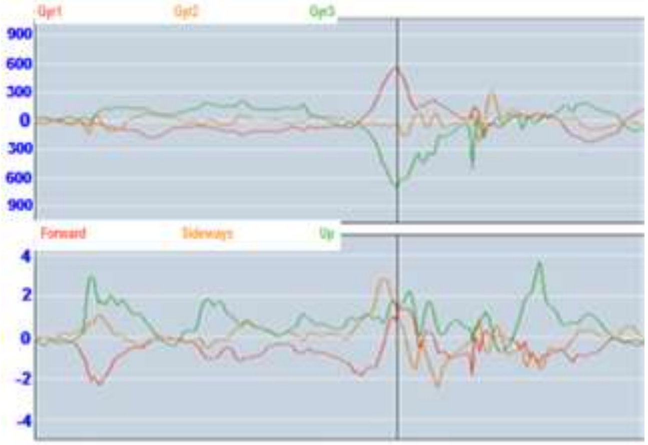


Figure 1: Gyroscope and accelerometer traces for backhand shot marked with the vertical line.

As our main feature, we calculated a feature called *Peak_strength* as follows:

- Calculate absolute sum of angular speeds on axes 1 and 3
- Raise it to the power 4, to emphasize higher values
- Apply Butterworth band-pass filter with high and low cutoff frequencies of 1.5 and 25 Hz.
- To get the final *Peak_strength* value, set the lower peak to zero when two peaks are too close (the distance was set by a domain expert to 1.3 s)

High *Peak_strength* values calculated in this way mark potential shots. Additionally, we calculated several other features. We set two different window sizes (0.8 s and 1.2 s) and calculated the average values, variances and standard deviations for each accelerometer and gyroscope axis. We added the sums of and differences between all pairs of gyroscope axis values and also between accelerometer axis values. We also calculated the speed of movement from the GPS coordinates. To illustrate its importance, Figure 2 shows how the combinations of *Peak_strength* and speed of movement separates shots and shot attempts (high *Peak_strength* values that are not shots).

3.2 Experimental Setup

We divided the evaluation in two parts. Firstly, we evaluated how well we can detect if a shot has occurred, and secondly, we tried to detect which type of shot was made.

For building the models, after empirical comparison of several algorithms, we chose the Random Forest (RF) [2] algorithm. Each RF consisted of 10 decision trees, the minimum number of samples required to split an internal node was 8, and the minimum number of samples required at a leaf node was 4.

We evaluated the models in two ways. Firstly, we performed 10-fold cross validation using Stratified shuffle split [3]. This procedure ensured equal class distributions between training and test sets. Secondly, we used the leave-one-player-out approach (LOPO), where we used one player's data for testing and the data from the other players for training. This approach enables us to estimate the accuracy of the models for previously unseen players with different shot techniques.

When evaluating the models, we classified each data entry (10 ms) as a shot or no-shot, and the type of shot. With this approach almost all the data points were classified as no-shots, so calculating the classification accuracy would be useless. We therefore focused on the precision and recall.

3.3 Results

The results for detecting shots and shot types for the cross-validation and for the LOPO approach are presented in Tables 1 and 2.

	Cross-validation	LOPO
Precision	97.3%	97.3%
Recall	96.6%	96.5%

Table 1: Precision and recall for detecting tennis shots

	Cross-validation			
	Foreh.	**Backh.**	**Serve**	**All**
Precision	95.3%	94.3%	99.1%	96.2%
Recall	91.4%	90.2%	99.3%	93.6%
	LOPO			
	Foreh.	**Backh.**	**Serve**	**All**
Precision	91.5%	93.6%	99.8%	95.0%
Recall	90.5%	90.6%	98.2%	93.1%

Table 2: Precision and recall for detecting types of tennis shots

As we can see, the precision and recall obtained with cross-validation and LOPO are very similar. This means that the built models are relatively independent from the type of player or his technique or style of play.

The main sources of errors are fast unnatural body rotation movements and special events that occur during the play. An example from our data set is a player warming up doing very similar body movements as during shots, or a player throwing his racquet at the fence with the same body movement as when serving.

4. DETECTING ACTIVE PLAY

The algorithm for detecting active play during a tennis game could only be developed after we have detected the shots. The reason is that we want our algorithm not only to have a high classification accuracy, but also to include as many shots as possible in the detected active play. In other words, misdetection of active play is less undesirable when no shots are made. So due to having two objectives, the detection was formulated as a multi-objective optimization problem.

4.1 Feature Extraction

The main idea when detecting the starting (end ending) point of a sequence of active play (rally) was that at this point the difference between the activity before and after would be the largest.

We used accelerometer values because they better represent the players' movement then gyroscope values, which primarily spike when making shots. From these values, we calculated a modified variance that gives more emphasis to the largest variations in data traces:

$$\text{var*} = \frac{\sum_{i=1}^{N}(x_i - \overline{x})^4}{N}$$

So for each data point, we calculated three additional features based on var*: the back overall variance (BV), the forward overall variance (FV) and the difference between these two (DV = BV – FV). FV and BV are calculated as the sum of the var* for each of the three acceleration axes, on the sequences immediately before (BV) and after (FV) a potential beginning or end of a rally (the size of the sequences was subject to optimization).

To be able to truly detect the best point describing the beginning or end of each rally, we also calculated peaks on the DV feature. Calculating the peaks was done the same way as for the shot detection. The minimum distance between peaks was subject to the optimization.

4.2 Problem Formulation

For each data point we calculated the previously described features, and set a rule for detecting the beginning of a rally and a rule for detecting the end of a rally. A data point is marked as the beginning of a rally if it satisfies the following rule:

$$(DV > p1)\ \&\ ((BV < p3)\ ||\ (FV < p4)).$$

A data point is marked as the end of the rally if it satisfies the following rule:

$$(-DV > p2)\ \&\ ((BV < p3)\ ||\ (FV < p4)),$$

where parameters $p1$, $p2$, $p3$, $p4$ were determined through optimization. Both rules consist of two parts. The first part determines the threshold for the change in activity before and after a potential beginning or end of a rally. For the beginning of a rally, this difference is usually larger because a rally often starts explosively and ends gradually, so the thresholds $p1$ and $p2$ can be different. The second part is the same for both rules and serves to remove false detections due to the variation in intensity during the rally by specifying that the activity, either before or after the beginning or end of a rally, should be low.

So altogether we optimized six input parameters: sequence size, minimum distance between peaks, $p1$, $p2$, $p3$ and $p4$.

4.3 Experimental Setup

To optimize the two objectives – classification error and the number of shots not inside the detected rallies – we used the well-known evolutionary multi-objective optimization algorithm called NSGA-II [4]. The population size was set to 25, the stopping criterion was set to 10,000 solution evaluations, and the tournament selection was used.

4.4 Results

The final front of the optimization can be seen in Figure 3. We can see a typical result for a multi-objective optimization problem, a non-dominated front showing a tradeoff between objectives. We can also see a knee on the front labeled with a circle. In this solution six shots are missed, since they occurred without the surrounding intense activity which accompanied other shots. An example is a player hitting the ball out of court after the rally finished. To include even such shots in the detected rallies, we would need to sacrifice a lot of classification accuracy.

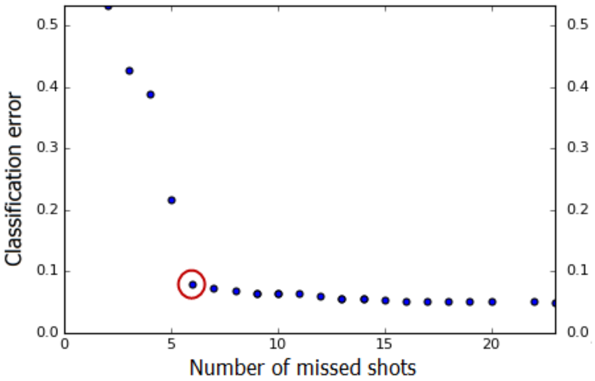


Figure 3: The final front showing the best solutions based on the classification error and the number of shots outside of detected active play.

Since our objective was to accurately detect the duration of the rallies, we chose one solution from the middle of the front and for this solution, we calculated the distribution of the durations of the rallies. The comparison with the manually labeled rallies can be seen in Figure 4.

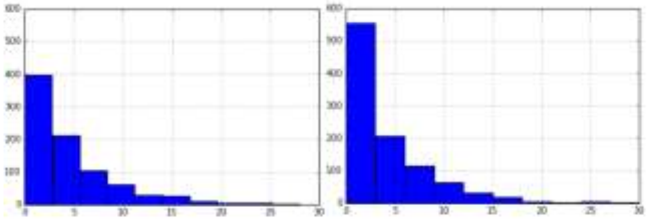


Figure 4: Comparing manually labeled (left) and automatically detected distributions for play durations.

We can clearly see the similarities between the distributions. The reason for the detected distribution having more very short rallies is that the algorithm detects even small starts of movement that we did not label as rallies because they were too short. For example, a server hitting the net with the first serve results in the returner making just a small movement.

By combining the classified shot types, detected active playing phases and locations from the GPS, we can calculate several useful metrics that help remove subjectivity from the game and allow for objective evaluation of different tactical approaches and training routines. An example of such a view can be seen in Figure 5, where we present the heat map of a player's position during active play and combine it with forehand and backhand shots as points of different color and size. We also included a dashed line that separates part of the court where more backhands are played from the one where more forehands were played. We can see that the player played more aggressively on the left side, thus his heat map is closer to the baseline. On the right side, a less aggressive approach allowed him to play more forehands and thus he dictated play by playing more often with his better shot.

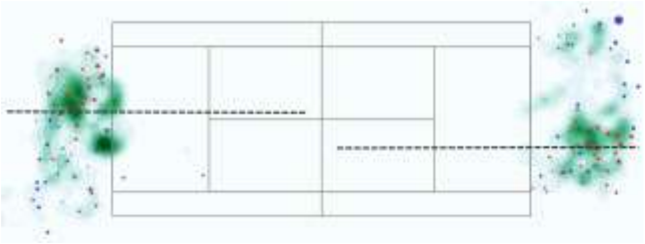


Figure 5: Heat map of a player's position during active play combined with shot locations (blue = forehand, red = backhand) and their *Peak_strength* (size of points).

5. CONCLUSION

In this article we presented a two-part algorithm for analyzing wearable sensor data for professional tennis players. Firstly, we detected and classified different shot types, and secondly, we distinguished the active playing phases from the time in-between points. By combining the procedures, players can get a unique perspective on their game which enables objective analysis in the tactical and physical sense.

For the future, we plan to equip both players with the same type of sensor, and by measuring the time difference between their shots and by calculating the distance between them, we will be able to calculate the average speed of ball and thus additionally quantify the quality of each shot.

6. ACKNOWLEDGMENTS

We would like to thank Catapult Sports for their sound advice and cooperation, and the national research agency (ARRS) for providing the research grant under the project PR-06884.

7. REFERENCES

[1] Renò, Vito, et al. "A technology platform for automatic high-level tennis game analysis." Computer Vision and Image Understanding (2017).

[2] Liaw, Andy, et al. "Classification and regression by RandomForest." R news 2.3 (2002): 18-22.

[3] Liberty, Edo, Kevin Lang, and Konstantin Shmakov. "Stratified sampling meets machine learning." International Conference on Machine Learning. 2016.

[4] Deb, Kalyanmoy, et al. "A fast and elitist multi-objective genetic algorithm: NSGA-II." IEEE transactions on evolutionary computation 6.2 (2002): 182-197.

[5] Yu, Xinguo, et al. "A trajectory-based ball detection and tracking algorithm in broadcast tennis video." Image Processing, 2004. ICIP'04. 2004 International Conference on. Vol. 2. IEEE, 2004.

[6] Pei, Weiping, et al. "An embedded 6-axis sensor based recognition for tennis stroke." IEEE International Conference on Consumer Electronics (ICCE), 2017.

[7] Yang, Disheng, et al. "TennisMaster: an IMU-based online serve performance evaluation system." Proceedings of the 8th Augmented Human International Conference. ACM, 2017

Anytime Benchmarking of Budget-Dependent Algorithms with the COCO Platform

Tea Tušar
Department of Intelligent Systems
Jožef Stefan Institute
Ljubljana, Slovenia
tea.tusar@ijs.si

Nikolaus Hansen
Inria
École Polytechnique CMAP (UMR 7641)
Palaiseau, France
nikolaus.hansen@inria.fr

Dimo Brockhoff
Inria
École Polytechnique CMAP (UMR 7641)
Palaiseau, France
dimo.brockhoff@inria.fr

ABSTRACT

Anytime performance assessment of black-box optimization algorithms assumes that the performance of an algorithm at a specific time does not depend on the total budget of function evaluations at its disposal. It therefore should not be used for benchmarking budget-depending algorithms, i.e., algorithms whose performance depends on the total budget of function evaluations, such as some surrogate-assisted or hybrid algorithms. This paper presents an anytime benchmarking approach suited for budget-depending algorithms. The approach is illustrated on a budget-dependent variant of the Differential Evolution algorithm.

1. INTRODUCTION

In black-box optimization, the problem to be optimized cannot be explicitly written as a function of its input parameters (if an underlying function exists, it is unknown). This is often the case with real-world problems where solutions are evaluated using simulations. Without the possibility of exploiting the structure of the function, optimization algorithms resort to repeatedly sample its decision space and use previously evaluated solutions to steer the search towards promising regions. Since the evaluations of real-world problem functions are often more time-consuming than the internal computations of optimization algorithms, the running time, or *runtime*, of an algorithm is generally measured by counting the number of performed function evaluations. The goal of an algorithm in black-box optimization is thus to find satisfactory solutions to the given problem in as few function evaluations as possible.

When measuring the performance of an algorithm in the black-box setting, we are interested in the required runtime to reach a target value. Or rather, we wish to obtain all runtimes corresponding to increasingly difficult targets [3]. In problems with a single objective, the targets are usually defined as differences from the optimal function value, while in problems with multiple objectives, the targets are determined as differences from the optimal value of a multiobjective performance indicator.

The proportion of reached targets plotted against the runtimes in the sense of an empirical cumulative distribution function yields a *data profile* [7]—a graph showing the *anytime* performance of an algorithm, essentially mimicking the convergence graph (the plot of best found function or indicator values over time). In addition to being easy to interpret, the data profile has another important advantage—it can be used to represent algorithm performance aggregated over multiple runs on different problems of the same dimensionality (see Section 2 for more details). This considerably alleviates presentation and understanding of algorithm results on a large number of problems.

The underlying assumption in anytime performance assessment is that the performance of an algorithm at a specific runtime does not depend on the total budget of function evaluations. That is, performance of an algorithm at 1000 function evaluations is expected to be the same if the algorithm was run with a budget of 1000 or 100 000 function evaluations (everything else being equal). If this is not the case, data profiles should not be employed to infer performance of the same algorithm with a total budget different from the one used in the experiments.

Algorithms can depend on the total budget for different reasons. Consider for example surrogate-assisted approaches. They construct surrogate models of the optimization problem and combine actual function evaluations with evaluations on the models. While some algorithms work in a budget-independent way, e.g. [6], others save some true function evaluations for the end (just before the budget is exhausted), making them budget-dependent, e.g. [9]. Similarly, hybrid genetic algorithms that combine genetic algorithms with local search methods can reserve a number of final function evaluations to additionally improve the current best solutions [2]. Another example of budget-dependent algorithms are evolutionary algorithms that set any of their parameters based on the total budget [1].

To address this issue, we propose an approach for benchmarking budget-dependent algorithms that allows anytime performance assessment of their results. It is based on the anytime benchmarking from the Comparing Continuous Optimizers (COCO) platform [4]. The approach is demonstrated on a budget-dependent variant of Differential Evolution (DE) [8] ran on the `bbob` test suite [5].

In the following, we first present some background on anytime benchmarking with the COCO platform (Section 2). The new approach is described in Section 3 followed by a discussion on its time complexity. An illustration with the DE algorithm is shown in Section 4. Section 5 presents some concluding remarks.

2. ANYTIME BENCHMARKING IN COCO

COCO (`https://github.com/numbbo/coco`) is a platform that facilitates benchmarking of optimization algorithms by automatizing this procedure and providing data of previously run algorithms for comparison [4]. An important part of COCO's anytime benchmarking approach [3] is the presentation of algorithm results in the form of data profiles [7].

Consider a single run of algorithm $\mathcal{A}$ on problem p. Given l increasingly difficult targets $\tau_1, \tau_2, \ldots, \tau_l$, it is easy to compute the corresponding runtimes $r_1^p, r_2^p, \ldots, r_l^p$ needed by algorithm $\mathcal{A}$ to reach each of these targets. If the target τ_j was not reached, r_j^p is undefined. A data profile for algorithm $\mathcal{A}$ is then constructed by plotting for each number of evaluations the proportion of targets reached by $\mathcal{A}$ in a runtime equal to or smaller than the number of evaluations. In other words, a data profile is the empirical cumulative distribution function of the recorded runtimes $r_1^p, r_2^p, \ldots, r_l^p$.

Data profiles can be further exploited to show aggregated information over randomized repetitions of running algorithm $\mathcal{A}$ on problem p. Instead of using repeated runs of $\mathcal{A}$ on p (which is sensible only for stochastic algorithms), randomization in COCO is achieved by running the same algorithm $\mathcal{A}$ on different instances of problem p (for example, translated versions of the same problem).

Consider k instances of the problem p, denoted here as $p(\theta_1)$, $p(\theta_2), \ldots, p(\theta_k)$. Like before, the runtime $r_j^{p(\theta_i)}$ at which algorithm $\mathcal{A}$ achieves target τ_j on problem instance $p(\theta_i)$ can be easily calculated for each i and j and is undefined when the target has not been reached. In order to be able to compare algorithms of different success probabilities (for example an algorithm that always reaches difficult targets, but does this slowly, with an algorithm that sometimes reaches a target quickly while other times fails to reach it at all), we simulate restarts of each algorithm via a *bootstrapping procedure*. The N bootstrapped simulated runtimes $r_{j,1}, r_{j,2}, \ldots, r_{j,N}$ of the artificially restarted algorithm to reach a target τ_j are computed from the *recorded* runtimes $r_j^{p(\theta_i)}$ of algorithm $\mathcal{A}$ (for a large N, e.g. $N = 1000$) as:

```
for $c \leftarrow 1, \ldots, N$ do                          ▷ Repeat $N$ times
    $r_{j,c} \leftarrow 0$                                   ▷ Initialize runtime
    loop
        $i \leftarrow$ random($\{1, \ldots, k\}$)            ▷ Choose an instance
        if $r_j^{p(\theta_i)}$ is defined then               ▷ Successful
            $r_{j,c} \leftarrow r_{j,c} + r_j^{p(\theta_i)}$
            break loop
        else                                                 ▷ Unsuccessful
            $r_{j,c} \leftarrow r_{j,c} + r_{\max}^{p(\theta_i)}$
        end if
    end loop
end for
return $r_{j,1}, r_{j,2}, \ldots, r_{j,N}$
```

if at least one of the recorded runtimes is finite. Note that the total runtime of $\mathcal{A}$ on $p(\theta_i)$, $r_{\max}^{p(\theta_i)}$, is added each time an unsuccessful trial is picked. Runtimes $r_{j,1}, r_{j,2}, \ldots, r_{j,N}$ are undefined for targets τ_j that were not reached in any of the problem instances. The resulting $N \cdot l$ runtimes (of which some undefined) are used to construct the data profile in an analogous way as before, but this time the y axis shows the proportion of targets reached out of $N \cdot l$ ones.

Finally, data profiles are also able to aggregate runtime results over problems of the same dimensionality that optimize a different function. Imagine a test suite consisting of m such problems with multiple instances. After bootstrapping is performed for each problem separately, there are $m \cdot N \cdot l$ function and target pairs and the same number of bootstrapped runtimes. The aggregated data profile for algorithm $\mathcal{A}$ can thus be constructed by plotting for each number of evaluations the proportion of function and target pairs reached by $\mathcal{A}$ in a runtime equal to or smaller than the number of evaluations.

It is important to note that runtimes are never aggregated over different dimensions since problem dimension is often used as an algorithm parameter. This also allows scalability studies. See [3] for more details on COCO's performance assessment procedure.

3. A BENCHMARKING APPROACH FOR BUDGET-DEPENDENT ALGORITHMS

The idea for benchmarking a budget-dependent algorithm $\mathcal{A}$ is very simple: the algorithm is run with increasing budgets and the resulting runtimes are presented in a single data profile. This is achieved by means of an 'artificial' algorithm $\widetilde{\mathcal{A}}$ that works as follows.

Consider K increasing budgets $b_1, b_2, \ldots, b_K$ and K budget-dependent algorithm variants $\mathcal{A}_{b_1}, \mathcal{A}_{b_2}, \ldots, \mathcal{A}_{b_K}$. The algorithm $\widetilde{\mathcal{A}}$ first works as algorithm $\mathcal{A}_{b_1}$ for budgets $b \leq b_1$, then works as algorithm $\mathcal{A}_{b_2}$ for budgets b, where $b_1 < b \leq b_2$, and so on, finishing by mimicking algorithm $\mathcal{A}_{b_K}$ for budgets b, where $b_{K-1} < b \leq b_K$ (see also Figure 1). In an algorithmic notation (where x_i denotes the ith solution explored by the corresponding algorithm):

```
$b_0 \leftarrow 0$                                  ▷ Initialize a budget preceding $b_1$
for $j \leftarrow 1, \ldots, K$ do                   ▷ Iterate over budgets
    for $i \leftarrow 1, \ldots, b_{j-1}$ do
        $\mathcal{A}_{b_j}(x_i)$                     ▷ Run $\mathcal{A}_{b_j}$ ignoring its output
    end for
    for $i \leftarrow b_{j-1} + 1, \ldots, b_j$ do
        $\widetilde{\mathcal{A}}(x_i) \leftarrow \mathcal{A}_{b_j}(x_i)$    ▷ $\widetilde{\mathcal{A}}$ mimics $\mathcal{A}_{b_j}$
    end for
end for
```

Although the first b_{j-1} evaluations of the algorithm $\mathcal{A}_{b_j}$ are ignored by $\widetilde{\mathcal{A}}$, they need to be performed so that $\mathcal{A}_{b_j}$ is in the correct state at evaluation $b_{j-1} + 1$, when it starts to be mimicked by algorithm $\widetilde{\mathcal{A}}$.

As shown with the red run in Figure 1, $\mathcal{A}_{b_j}$ might not contribute to $\widetilde{\mathcal{A}}$ at all if the performance of $\mathcal{A}_{b_i}$ is better for some $b_i < b_j$. On the other hand, if $\mathcal{A}_{b_j}$ is significantly better than $\mathcal{A}_{b_{j-1}}$ (for example, the green vs. the yellow run), this causes a 'jump' in the performance of $\widetilde{\mathcal{A}}$. Note also that the best-so-far profile of $\widetilde{\mathcal{A}}$ does not necessarily follow the overall best-so-far profile of $\mathcal{A}_{b_j}$, but only its best-so-far profile after b_{j-1} function evaluations (notice the yellow and

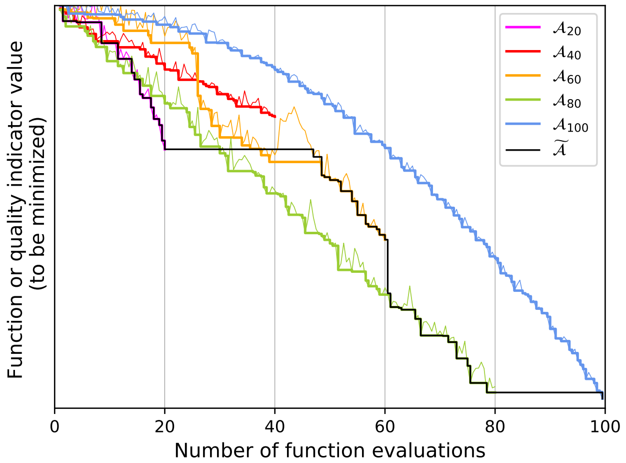


Figure 1: An illustration of the 'artificial' algorithm $\widetilde{\mathcal{A}}$ constructed from five runs of algorithm $\mathcal{A}$ ($\mathcal{A}_b$ means the algorithm was run with the budget of b function evaluations). Thin and thick lines show the actual and best-so-far performance for each run of $\mathcal{A}$, respectively.

black lines after 40 function evaluations).

Composing the performances of algorithm $\mathcal{A}$ with different budgets into algorithm $\widetilde{\mathcal{A}}$ results in an estimation of the anytime performance of $\mathcal{A}$. The quality of the estimation depends on the number of budgets K—more budgets enable a better estimation, but make the procedure more time consuming.

One could run the budget-dependent variants of algorithm $\mathcal{A}$ for every budget between 1 and b_K thus obtaining the best possible estimate. However, this would require

$$\sum_{j=1}^{b_K} j = \frac{b_K(b_K+1)}{2}$$

total evaluations. Diluting the budgets by taking only every Mth one does not help to significantly reduce the number of total evaluations. A more promising approach is that of using equidistant budgets in the logarithmic scale. For example, K such budgets between 1 and 10^M require

$$\sum_{j=0}^{MK} 10^{j/K} = \frac{10^{1/K+M}-1}{10^{1/K}-1}$$

total evaluations. Table 1 contains total evaluations for this case for some values of K and M. The actual number of evaluations is likely to be smaller than these numbers due to some consecutive (small) budgets being rounded to the same integer number.

4. EXAMPLE

We present a small example to demonstrate the proposed anytime benchmarking procedure on the COCO platform. The algorithm used in this example is a budget-dependent variant of Differential Evolution (DE) [8], a well-known evolutionary algorithm. While the original DE algorithm is budget-independent, a study shows that setting its parameters, especially population size, in connection to the total budget of evaluations can improve its results [1].

Table 1: An upper bound of function evaluations required for benchmarking budget-dependent algorithms with K budgets between 1 and 10^M that are equidistant in the logarithmic scale, for some selected values of K and M.

$\left\lceil\frac{10^{1/K+M}-1}{10^{1/K}-1}\right\rceil$		M		
		3	4	5
K	10	4.859	48.618	486.208
	20	9.188	91.947	919.540
	50	22.198	222.165	2.221.835
	100	43.889	439.270	4.393.094

Table 2: Population size of the budget-dependent DE computed for some selected values of budget multipliers m_{budg} and problem dimensions n.

$\lfloor 3\log_{10}^2(n \cdot m_{\text{budg}})\rfloor$		m_{budg}		
		10	100	1000
n	2	5	15	32
	5	8	21	41
	10	12	27	48
	20	15	32	55

In the experiments we use the DE implementation from the `scipy` Python package (`https://www.scipy.org/`) with the following parameters:

- Initialization = Latin Hypercube sampling
- DE strategy = best/1/bin
- Population size = *variable* (see text)
- Weight F = random in the interval $[0.5, 1)$
- Crossover probability $CR = 0.7$
- No local optimization of the final solution
- Relative tolerance for convergence = 10^{-9}

This implementation computes the population size based on the problem dimension n (for a user-specified multiplier m_{pop}, the population size is calculated as $n \cdot m_{\text{pop}}$). This was bypassed in order to make population size budget-dependent. For a problem with n dimensions and a budget multiplier m_{budg}, the actual budget in COCO is computed as $n \cdot m_{\text{budg}}$ and the population size of DE is calculated as

$$\lfloor 3\log_{10}^2(n \cdot m_{\text{budg}})\rfloor .$$

Table 2 gathers the values of this formula for selected budget multipliers m_{budg} and problem dimensions n.

The experiment consisted of running five instances of DE with budget multipliers from $\{10, 31, 100, 316, 1000\}$ and at the same time composing their performances into the anytime artificial algorithm called 'DE-anytime'. All algorithms were run on the 24 problems functions from the `bbob` test suite [5] with dimensions n in $\{2, 3, 5, 10, 20\}$. Each problem was instantiated 15 times. The results for dimension 10 are presented in Figure 2. In data profiles plots produced by

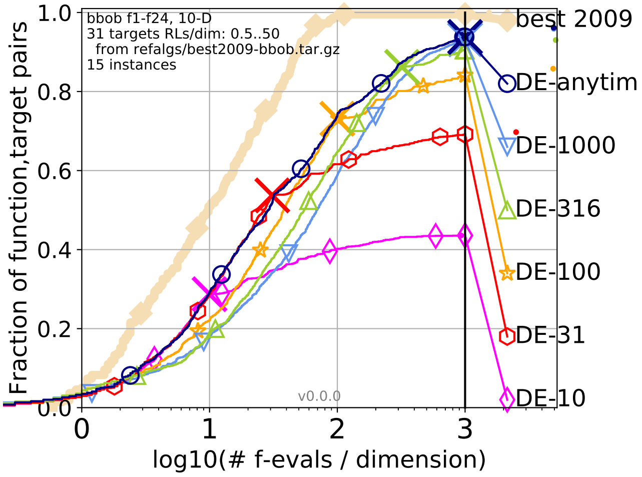


Figure 2: Data profiles for budget-dependent variants of DE run with budgets of $\{10, 31, 100, 316, 1000\} \cdot n$ number of evaluations and the 'artificial' algorithm constructed from these variants estimating anytime performance of DE. See text for further information.

COCO, the function evaluations are always divided by the problem dimension n and shown on a logarithmic scale.

The benchmark setting used in this example is COCO's *expensive setting*, in which the number of evaluations is limited to 1.000n and the 31 targets are defined in a relative way—according to the performance of a virtual algorithm denoted with 'best 2009' that is comprised of the best results achieved by 31 algorithms at the Black-box Optimization Benchmarking (BBOB) workshop in 2009. The targets are chosen from $[10^{-8}, \infty)$ such that the 'best 2009' algorithm just failed to reach them within the given budget of nm_{budg} evaluations, with 31 different values of m_{budg} chosen equidistantly in logarithmic scale between 0.5 and 50.

We are showing the results for dimension 10, since the differences among DE instances are best visible for this dimension. Note that the algorithms were stopped at the moment denoted by the large cross, but the data profiles increase also beyond that point due to bootstrapping (see Section 2).

From Figure 2 we can observe that DE variants with a budget of 10n and 31n evaluations achieve a very similar performance in the first 10n evaluations. Other DE variants are noticeably different from the first two and also among themselves, with those with lower budgets converging faster at the beginning of the run. This confirms the findings from [1] that better performance can be achieved by fitting the population size to the total budget.

The dark blue data profile corresponding to the 'DE-anytime' artificial algorithm follows the five underlying algorithms as expected. The accuracy of this estimate could be further improved if a higher number of different budgets was used.

5. CONCLUSIONS

We presented a novel approach for benchmarking budget-dependent algorithms that enables anytime performance assessment of their results. The approach demands repeated runs of an algorithm with increasing budgets. Depending on the number and size of these budgets, it can take a significant amount of time (it is quadratic in the maximal budget in the worst case). By using budgets that are equidistant in the logarithmic scale, the time complexity depends linearly on the maximal budget, making the approach more usable in practice. An example experiment showing how to use this approach in COCO will be available in COCO v2.2.

6. ACKNOWLEDGMENTS

The authors wish to thank Joshua Knowles for helpful discussions on this topic during the SAMCO workshop (Workshop on Surrogate-Assisted Multicriteria Optimization) at the Lorenz Center in Leiden, The Netherlands, 2016.

The authors acknowledge the financial support from the Slovenian Research Agency (project No. Z2-8177). This work is part of a project that has received funding from the *European Union's Horizon 2020 research and innovation program* under grant agreement No. 692286.

This work was furthermore supported by a public grant as part of the Investissement d'avenir project, reference ANR-11-LABX-0056-LMH, LabEx LMH, in a joint call with Gaspard Monge Program for optimization, operations research and their interactions with data sciences.

7. REFERENCES

[1] A. S. D. Dymond, A. P. Engelbrecht, and P. S. Heyns. The sensitivity of single objective optimization algorithm control parameter values under different computational constraints. In *Proceedings of CEC 2011*, pages 1412–1419, 2011.

[2] T. A. El-Mihoub, A. A. Hopgood, L. Nolle, and A. Battersby. Hybrid genetic algorithms: A review. *Engineering Letters*, 13:124–137, 2006.

[3] N. Hansen, A. Auger, D. Brockhoff, D. Tušar, and T. Tušar. COCO: Performance assessment. *ArXiv e-prints*, arXiv:1605.03560, 2016.

[4] N. Hansen, A. Auger, O. Mersmann, T. Tušar, and D. Brockhoff. COCO: A platform for comparing continuous optimizers in a black-box setting. *ArXiv e-prints*, arXiv:1603.08785, 2016.

[5] N. Hansen, S. Finck, R. Ros, and A. Auger. Real-parameter black-box optimization benchmarking 2009: Noiseless functions definitions. Research Report RR-6829, INRIA, 2009.

[6] D. R. Jones, M. Schonlau, and W. J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998.

[7] J. J. Moré and S. M. Wild. Benchmarking derivative-free optimization algorithms. *SIAM Journal on Optimization*, 20(1):172–191, 2009.

[8] R. Storn and K. Price. Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):341–359, 1997.

[9] V. Volz, G. Rudolph, and B. Naujoks. Surrogate-assisted partial order-based evolutionary optimisation. In *Proceedings of EMO 2017*, pages 639–653, 2017.

Criteria for Co-existence of GM and Conventional Maize Production

Marko Debeljak
Jožef Stefan Institute and
Jožef Stefan International
Postgraduate School
Jamova 39, Ljubljana,
Slovenia
+386 1 477 3124
marko.debeljak@ijs.si

Florence Leprince
ARVALIS-Institut du vegetal,
21, chemin de Pau,
Montardon, France
+33 5 59 12 67 79
f.leprince@arvalisinstitutduvegetal.fr

Sašo Džeroski
Jožef Stefan Institute and
Jožef Stefan International
Postgraduate School
Jamova 39, Ljubljana,
Slovenia
+386 1 477 3217
saso.dzeroski@ijs.si

Aneta Trajanov
Jožef Stefan Institute and
Jožef Stefan International
Postgraduate School
Jamova 39, Ljubljana,
Slovenia
+386 1 477 3662
aneta.trajanov@ijs.si

ABSTRACT

The criteria for co-existence of genetically-modified (GM) and conventional (non-GM) crops must reflect the best available scientific evidence on mixture between these two types of crops. Co-existence strategies based on fixed isolation distances are not in line with the EC guidelines on co-existence, which require criteria adaptable to local constraints. In this paper, we apply data mining for identification of co-existence criteria of maize production. We use classification trees to generate co-existence criteria for GM and conventional maize fields. The data used in this study were provided by ARVALIS and consisted of several surveys of outcrossing between pairs of maize fields. Based on the model structure, the most important co-existence criteria are flowering time lag, wind direction, presence of isolation rows and distance between the GM and conventional field. The co-existence criteria generated from the model for prediction of outcrossing were applied on an independent Spanish dataset. The results are meaningful and in accordance with literature and have high potential for application in the development of computer based co-existence decision support system.

Keywords

Genetically modified crops, GM maize, co-existence criteria, classification trees, random forests.

1. INTRODUCTION

Co-existence is concerned with the potential economic impact of the mixture of genetically modified (GM) and non-GM crops, the identification of workable management measures to minimize mixture, and the cost of these measures [5]. Co-existence applies only to approved GM crops that were considered to be safe prior to their commercial release and safety issues fell outside its remit [14]. EC regulation 1829/2003 (article 43) [6] provides guidelines to develop national or regional strategies and best practices to ensure co-existence. However, the selection of preventive co-existence criteria is the individual responsibility of each Member State.

The level of purity needed to ensure co-existence is defined by a tolerance threshold. The EU accepts an adventitious or technically unavoidable presence of authorized GM material in non-GM food and feed up to 0.9% and the main task of co-existence is to find out by which means the adventitious presence can be kept below the accepted threshold level. In particular, the prediction of adventitious presence of GM material in neighboring non GM fields is required in order i) to assess the co-existence performance of applied management strategies in GM fields, and ii) to identify efficient crop management measures that enable the co-existence of GM and conventional crop production systems.

The identification of co-existence criteria is currently based on two approaches. The first uses a mechanistic matrix modeling approach that is based on a theoretical description of pollen dispersal, while data from field experiments are used for calibrations and validations of such models [1]. The second approach is based on empirical knowledge about co-existence, obtained mostly from observations and experiences from growing GM crops under real production conditions, where the performance of fixed co-existence measures is used and evaluated. Such an empirical model, called a global index, has been developed by [11].

In this study, we propose a third approach that employs techniques of data mining to real data about cross-pollination between GM and conventional crops grown under different crop management practices. Our goal is to identify co-existence criteria about the adventitious presence of GM maize in the conventional maize production, using the official threshold level of 0.9%, from the structure of the induced predictive models built from real data.

2. MATERIALS AND METHODS

2.1 Data

In this study, we used data provided by ARVALIS - Institut du végétal, France, and Institut de Recerca i Tecnologia Agroalimentàries (IRTA), Spain. The data provided by ARVALIS were used for construction of the data mining models and the Spanish data were used for validation of the induced co-existence criteria.

The data provided by ARVALIS were from surveys of outcrossing (gene flow from GM donor crop to recipient non-GM crop) between pairs of 88 maize fields in the Pau and Toulouse regions in the South – West of France, in the period from 2001-2007. Each field was described with the following variables: location of the fields, distances between donor and recipient fields, locations of sampling points, number of border rows, field size, sowing date, flowering date, flowering time-lag between donor and recipient fields, isolation distance between pairs of donor and recipient fields, prevailing wind direction from donor fields during the flowering period, and the percentage of outcrossing (outcrossing rate) in recipient fields using real-time quantification system-polymerase chain reaction.

The Spanish data were collected at harvesting time on 13 non-GM maize fields in 400 ha large maize crop area Pla de Foixà region (Girona), in Catalonia, Spain, in 2004, 2005 and 2006. During these years, the crop type, variety, sowing and flowering dates of maize fields were recorded, as well as meteorological data. At harvesting time, samples were collected on non-GM maize fields (7 fields in 2004, 4 fields in 2005 and 4 fields in 2006). The samples were analyzed by RTQ-PCR to evaluate the cross-pollination between GM and conventional maize [11]. The spatial distribution of crops in the selected region was described by maps generated from aerial photographs.

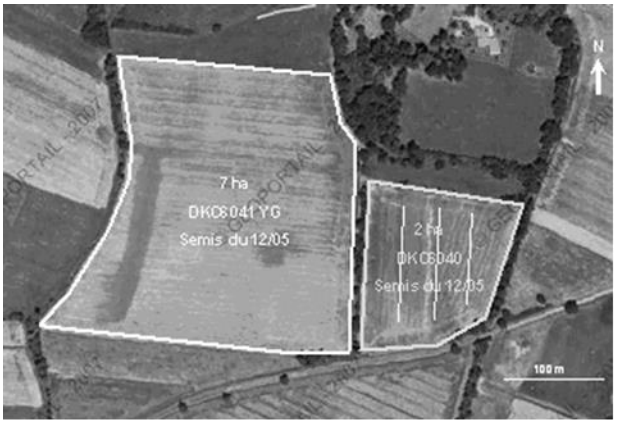


Figure 1: Example of the field-to-field map where the donor maize field is on the left and the recipient maize field is on the right side. Sampling points in the recipient maize field are indicated with lines.

2.2 Data mining methods

To find interactions between the attributes describing geographical, environmental and management parameters and outcrossing rates measured at sampled fields, we used data mining methods for induction of decision trees, which are ideally suited for analysis of complex ecological data. Decision trees predict the value of a dependent variable (target) from a set of independent variables (attributes). In our case, the dependent (target) variable is the outcrossing rate between a GM and a conventional maize in the field, which can have two values: 1 (below the threshold of 0.9% of adventitious presence) or 2 (above the threshold of 0.9% of adventitious presence). In this study, we used classification trees to develop predictive models for co-existence of GM maize production.

To evaluate the induced data mining models, we used two measures of performance or agreement of the discrepancy between measurements and predictions. We first calculated classifier's accuracy as the proportion of samples for which the category (below threshold vs. above threshold) was correctly predicted. We used 10-fold cross-validation as the most common and standard way of estimating the performance (accuracy) of a learning technique on unseen cases [15].

Data from real field studies describe the actual practices applied on the fields, where in the case of growing GM crops, incorporation of precautions for prevention of outcrossing of GM material to conventional fields is obligatory. Therefore, most often, these data are highly imbalanced, having a low number of outcrossing events. In our case, most of the samples (around 90%) were below the threshold of 0.9% of adventitious presence of GM material, while only around 10% were above this threshold. In cases like this, the accuracy is not the optimal performance metric for evaluation of the data mining models. Therefore, we used an additional performance metric - the Area Under the Receiver-Operator characteristic Curve (AUROC) [7], to more objectively evaluate the performance of the models. AUROC is defined for binary classification, where one of the classes is considered positive. A discrete classifier produces a pair of False Positives Rate (FPR: negatives incorrectly classified / total negatives) and True Positives Rate (TPR: positives correctly classified / total positives), which corresponds to a single point in the ROC space, while classifiers that return a probability value for the positive class correspond to an ROC curve. AUROC values of 0.7 and higher are considered to indicate a good fit to the data [7].

2.3 Data preprocessing

We used outcrossing grains as a unit of the outcrossing rate for all samples and surveys given in percentage of DNA [10].

From the original data described in section 2.1, we calculated the following attributes that we later used in the data mining analyses:

- Minimal distance from a sampling point to the donor field [m]
- Isolation distance (minimal distance between the donor and recipient field) [m]
- Presence of isolation rows [yes, no]
- Wind direction [0 – upwind (from recipient field), 1 – downwind (to recipient field)]
- Flowering time-lag [minimal (0-7 days), medium (8-14 days) and large (more than 15 days)]
- Common border length between donor and recipient field [m]
- Outcrossing rate [1 - if < 0.9% GM grains, 2 - if ≥ 0.9% GM grains]

We discretized some of the initial attributes in order to obtain comprehensible and easily interpretable predictive models of outcrossing. The ranking and threshold values for *Flowering time-lag* attribute were selected according to expert knowledge about maize production, while the target variable *Outcrossing rate* was discretized according to the accepted European threshold (0.9 % grains).

To deal with the high imbalance in the data, we applied methods, such as up-sampling and down-sampling of the dataset, in order to create a more balanced dataset. However, the newly obtained balanced dataset did not improve the results, so we stayed with the original dataset.

The Spanish data had a different structure than the French dataset, so to use it for validation of the generated co-existence criteria, we standardized the data to achieve the same dataset structure as in the case of the ARVALIS dataset used for building data mining models (Section 3.1). First, we unified the units of the measurements on the fields. Then, we reorganized and calculated the data for pair-based comparisons of donor-recipient fields to create a setting similar to the setting in the ARVALIS experiments.

3. RESULTS

3.1 Data mining models

To generate classification trees, we used the algorithm J4.8, an implementation of the C4.5 algorithm within the WEKA suite [15].

The classification model constructed on training data (Fig. 2) correctly classified 94.31% of the instances, while the cross-validated model correctly classified 90.24% of the instances (Table 1). However, the accuracy is sensitive to imbalanced data and therefore, the model correctly classifies most of the instances belonging to class 1, but it misclassifies most of the instances belonging to class 2.

The most unbiased measure of the goodness of a model is its AUROC value. The AUROC values of the classification trees obtained on training data and with cross-validation are given in Table 1. The AUROC value of the classification tree obtained with cross-validation is (0.576). This value is very close to 0.5 (the diagonal y = x), which means that the predictive power of the classification tree obtained with cross-validation is not very high. The predictive power of the classification tree obtained on training data according to its AUROC value (0.850) is much higher and indicates good predictive power.

Table 1: Accuracy and AUROC values for the J48 algorithm applied to the ARVALIS data.

	Accuracy	AUROC value
J48 on training data	94.31%	0.850
J48 10-fold cross-validation	90.24%	0.576

3.2 Co-existence criteria

The predictive power of the obtained classification tree (Fig. 2) is not very high due to the imbalanced data, therefore it cannot be used for making predictions about the outcrossing between GM and non-GM fields. However, its descriptive power is very relevant. The topmost part of the classification tree includes the attributes time-lag, wind direction, isolation distance and presence of isolation rows. Because of their dominant position in the model structure, they can be recognized as the ones which play the most important role in the cross-pollination process and as such they could present the most important criteria for co-existence of GM and conventional maize production.

Experts from ARVALIS and IRTA, as well as extensive literature [4,9,11,12,13] confirm the importance of the attributes captured in the model structure for the outcrossing process. Therefore, we focused on the parts of the model that correctly predicts outcrossing below 0.9% (leaves in the tree that provide most accurate predictions, presented by the number of examples that fall in a leaf and the number of incorrectly classified examples in Figure 2) and arranged our findings into a coherent and consistent list of co-existence criteria (Table 2).

In order to obtain robust and applicable co-existence criteria, we validated them on an independent dataset provided by IRTA, Spain.

The co-existence criteria (Table 2) derived from the classification tree generated from the French data, were confirmed by the Spanish data in 88.6% cases. Validation of the individual criterions (rules) has shown that the criterion 1 (time-lag) was confirmed in 90% of the cases, 76% of cases confirmed criterion 2 (combination of time-lag and wind direction), while criterion 4 was valid for 100% of the cases. None of the surveyed Spanish maize fields had protection rows, therefore, we were not able to validate criterion 3.

The validation of the outcrossing model (Figure 2) and the co-existence criteria, confirmed that there is a high potential for their application in the assessment process of co-existence issues.

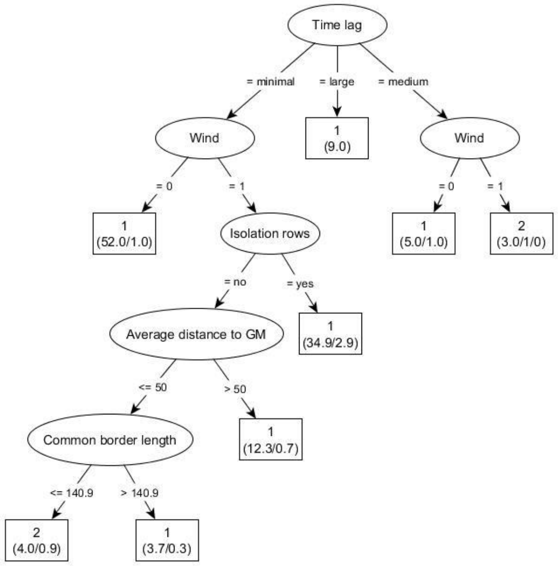


Figure 2: Outcrossing model induced by J48 (values in the leaves: 1: outcrossing < 0.9% grains, 2: outcrossing ≥ 0.9% grains).

Table 2: Proposed co-existence criteria for GM and conventional maize production.

1.	If the flowering time-lag is ≥ 15 days, then the outcrossing rate in a recipient field is below 0.9%.
2.	If the flowering time-lag is less than 15 days and if the prevailing wind direction during flowering period is from recipient to donor field, the outcrossing rate of a recipient field is below 0.9%.
3.	If the time-lag is less than 15 days and the wind direction during flowering period is from donor to recipient field, but the donor or the recipient filed has an isolation or protection row, then the outcrossing rate in the recipient field is below 0.9%.
4.	If the time-lag is less than 15 days and wind direction during flowering days is from donor to recipient field, but there are no isolation nor protection rows and the distance between the donor and the recipient field is more than 50 m, then the outcrossing rate in the recipient field is below 0.9%.

4. CONCLUSIONS

Compared to most of the studies about maize cross-fertilization due to pollen flow, we used the opportunity to gain new knowledge about this phenomenon by applying data mining techniques to explore the information stored in datasets of real maize growing management. This allowed us to overpass some shortages of previous field experimental designs that were mostly oriented toward worst-case scenarios (e.g., small donor field placed in the center of large recipient field [2,3] or spatial arrangements and distribution of donor and recipient experimental fields that were too simplified compared to the real ones [8].

Data describing real field experiments that involve GM crops are in general very imbalanced, due to the fact that farmers are obliged to take measures to prevent or minimize the outcrossing from GM to conventional fields. In addition, field experiments about outcrossing rates show a fast decrease of outcrossing by a distance from donor field [13]. Because of these reasons, the datasets we used contained much larger number of sampling points with outcrossing below 0.9%, which resulted in a very imbalanced structure of the data.

To mitigate this problem we studied different performance measures to assess the goodness of the obtained models. These measures showed that the models are very precise in predicting the situations when outcrossing is less than 0.9%, but not that precise when predicting the situations when the outcrossing is above 0.9%. Therefore, we accepted a compromise to make a recommendation about the co-existence criteria that are taken from the structure of the predictive outcrossing model. However, the part of the model describing conditions, which lead to the outcrossing rate above 0.9% is not very reliable.

Our study made a significant progress about using data mining methods for identification of co-existence criteria. However, the imbalance of the data is a problem that needs to be addressed by applying data mining methods that are suited for analyzing that kind of data, such as cost-sensitive learning. Finally, in this study, we were focused on the pair-based effects of outcrossing between GM and non-GM fields. To assess the multi-field effects on the outcrossing rate at a selected recipient field, a different data mining setting should be created and other data mining methods can be exploited, such as methods for multi-target prediction and inductive logic programming. Furthermore, this study shows the ability of data mining methods to extract useful information about co-existence issues from data describing real and not experimental maize production settings. By that, data mining models for prediction of outcrossing under real crop production conditions could be successfully incorporated in a computer based co-existence decision support system.

5. REFERENCES

[1] Beckie, H.J., Hall, L.M. 2008. Simple to complex: Modelling crop pollen-mediated gene flow. *Plant Sci* 175, 615-628.

[2] Belcher, K., Nolan, J., Phillips, P.W.B. 2005. Genetically modified crops and agricultural landscapes: spatial patterns of contamination. *Ecol Econ* 53, 387– 401.

[3] Debeljak M., Demšar D., Džeroski S., Schiemann J., Wilhelm R., Meier-Bethke S. 2005. Modeling outcrossing of transgenes in maize between neighboring maize fields. In: *Proceedings of the 19th International Conference Informatics for Environmental Protection (EnviroInfo)* (Hrebicek J., Jaroslav R. eds.). Brno, Czech Republic. pp. 610–614.

[4] Della Porta, G., Ederle, D., Bucchini, L., Prandi, M., Verderio, A., Pozzi, C. 2008. Maize pollen mediated gene flow in the Po valley (Italy): source-recipient distance and effect of flowering time. *Eur J Agron* 28, 255–265.

[5] European Commission, 2003a. Commission recommendation on guidelines for the development of national strategies and best practices to ensure the coexistence of genetically modified crops with conventional and organic farming. *Off J Eur Union L* 189, 36-47.

[6] European Commission, 2003b. EC regulation no. 1829/2003 of the European Parliament and of the council of 22 September 2003 on genetically modified food and feed. *Off J Eur Union L* 268, 1-23.

[7] Fawcett, T. 2006. An introduction to ROC analysis. *Pattern Recogn Lett* 27, 861-874.

[8] Loos, C., Seppelt, R., Meier-Bethke, S., Schiemann, J., Richter, O. 2003. Spatially explicit modelling of transgenic maize pollen dispersal and cross-pollination. *J Theor Biol* 225, 241–255.

[9] Meier-Bethke, S., Schiemann, J. 2003. Effect of varying distances and intervening maize fields on outcrossing rates of transgenic maize. In: *1st European Conference on the Co-existence of Genetically Modified Crops with Conventional and Organic Crops* (Boelt B. ed.). Research Center Flakkebjerg, pp. 77–78.

[10] Messéan, A., Angevin, F., Gomez-Barbero, M., Menrad, K., Rodriguez Cerezo, E., 2006. New case studies on the coexistence of GM and non-GM crops in European Agriculture. Technical Report Series of the Joint Research Center of the European Commission. Institute for Prospective Technological Studies, Sevilla.

[11] Messeguer, J., Pеńas, G., Ballester, J., Bas, M., Serra, J., Salvia, J., Palaudelmàs, M., Melé, E. 2006. Pollen-mediated gene flow in maize in real situations of coexistence. *Plant Biotechnol J* 4, 633–645.

[12] Palaudelmàs, M., Messeguer, J., Peñas, G., Serra, J., Salvia, J., Pla, M., Nadal, A., Melé, E. 2007. Effect of sowing and flowering dates on maize gene flow. In: *Book of abstracts of the third International Conference on Coexistence between Genetically Modified (GM) and non-GM-based Agricultural Supply Chains* (Stein A.J., Rodríguez-Cerezo E. eds.). EC, pp. 235–236.

[13] Sanvido, O., Widmer, F., Winzlerm, M., Streitm, B., Szerencsits, E., Bigler, F. 2008. Definition and feasibility of isolation distances for transgenic maize. *Transgenic Res* 17, 317-355.

[14] Schiemann, J. 2003. Co-existence of genetically modified crops with conventional and organic farming. *Environ Biosafety Res* 2, 213-217.

[15] Witten, I.H., Frank, E. 2011. Data Mining: Practical Machine Learning Tools and Techniques - 3rd edition. Morgan Kaufmann.

Knowledge Discovery from Complex Ecological Data: Exploring *Syrphidae* Species in Agricultural Landscapes

Marko Debeljak
Jožef Stefan Institute and
Jožef Stefan International
Postgraduate School
Jamova 39, Ljubljana,
Slovenia
+386 1 477 3124
marko.debeljak@ijs.si

Vladimir Kuzmanovski
Jožef Stefan Institute and
Jožef Stefan International
Postgraduate School
Jamova 39, Ljubljana,
Slovenia
+386 1 477 3143
vladimir.kuzmanovski@ijs.si

Veronique Tosser
ARVALIS-Institut du vegetal,
91 720 Boigneville,
France
+33 1 64 99 23 15
v.tosser@arvalis.fr

Aneta Trajanov
Jožef Stefan Institute and
Jožef Stefan International
Postgraduate School
Jamova 39, Ljubljana,
Slovenia
+386 1 477 3662
aneta.trajanov@ijs.si

ABSTRACT

Modelling structures and processes in ecosystems, such as biodiversity has always been a complex task. Most often, ecological empirical data are incomplete, inconsistent, imbalanced or very complex and a lot of effort should be put into preprocessing of such data in order to carry out meaningful analyses and modelling.

In this study, we are dealing with biological pest control in agricultural landscapes. The improvement of the natural regulation of organisms detrimental to agricultural production through biological pest control has the potential to reduce the use of pesticides and has a positive impact on crop production and environment. Our research focuses on auxiliary species of the family *Syrphidae*, which control Aphid pest species. In particular, our goal is to describe taxonomical and functional diversity of syrphid species to assess the potential performance of biological pest control in the studied area.

In this paper, we present the extensive process of data preprocessing for the purpose of modelling the taxonomic and functional aspects of syrphid species.

Keywords

Knowledge-discovery, data preprocessing, *Syrphidae* species, taxonomic and functional diversity, landscape structure, data mining.

1. INTRODUCTION

Knowledge discovery from ecological data is becoming an increasingly complex task. One reason for this is that ecosystems are very complex by themselves, representing networks of interactions and interdependencies among its elements and the environment, which are most often difficult to describe, explain or measure.

Empirical ecological data add another dimension to this complexity. Namely, we are often faced with empirical ecological data that are incomplete, inconsistent, containing out-of-range values, collected at different temporal and spatial scales, dispersed in different databases, noisy and imbalanced [3]. Therefore, special attention should be paid on the preprocessing and quality check of ecological data in order to obtain meaningful results, which makes the data preprocessing a very central step in the knowledge discovery process [7].

In this study, we are dealing with the assessment of potential performance of biological pest control in agricultural fields in the Boigneville area (Central France). In particular, we are exploring the biodiversity of the auxiliary species of the family *Syrphidae* (hoverflies), which are the major predators of Aphid pest species. The enhancement of biological pest control in agricultural fields helps reduce the use of pesticides and has an overall positive impact on crop production and quality of the environment.

Our goal is to analyse and model the taxonomic and functional diversity of syrphid species in order to estimate their potential performance of pest control in the studied area.

Quantifying and assessing the biological diversity of empirical data collected in the fields is a non-trivial task and involves calculations of different diversity measures, such as Hill numbers, Shannon’s and Simpson’s indexes, as well as quality checking and transformations of the available data in order to address the problem of modelling the taxonomic and functional diversity of species. In addition, the available empirical data did not contain sufficient information needed for these calculations and transformations. Therefore, we needed to extract additional data from several different datasets in order to cover all needed aspects of biodiversity.

These calculations, data transformation and collation, are a necessary step towards preparing a high quality dataset, which will enable us to obtain meaningful results and models. In this paper, we present the complex data preprocessing task that was carried out in order to explore and describe the taxonomic and functional diversity of syrphid species.

2. MATERIALS AND METHODS

2.1 Data

The empirical data were provided by ARVALIS, Institut du vegetal. Data come from Boigneville and were collected in 2009, 2010 and 2011. Samples of syrphid species were collected with five Malaise and eight cornet traps (Figure 1) on a weekly basis between March and November. Samples of caught insects have been determined to the species level and number of caught specimens per species was counted.

To describe the ecological functional traits of syrphid species in Boigneville, we obtained an additional and extensive database “Syrph The Net” [6]. It includes coded information on species’ macrohabitats, microsites, traits, range and status. The database is updated annually it is used to analyse recorded species assemblages in relation to their habitat associations.

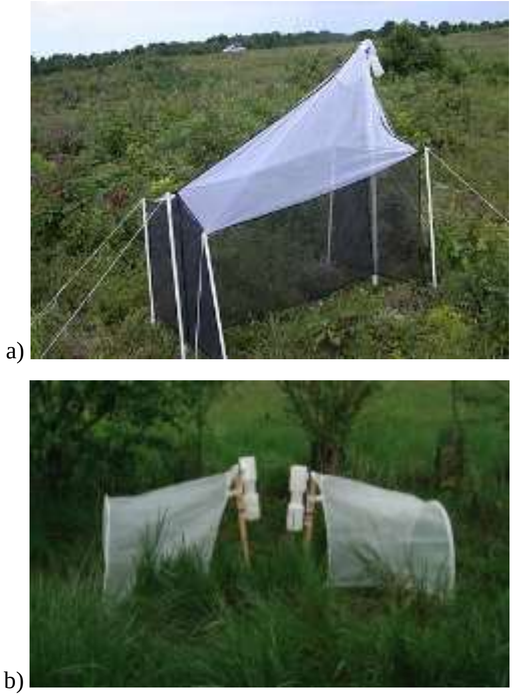


Figure 1: Malaise (a) and cornet (b) traps for sampling syrphid species

Data about landscape structure and crop properties were described from the original dataset for a 500 m and 1000 m radius around the traps (Figure 2). In delineated area, the surface of crops, natural vegetation (forests) and the length of linear corridors (tree lines, grass strips, grass pathways, hedges, roads) have been measured using GIS maps. Crop development stages were estimated for each crop in the studied area. Landscape description includes absolute and relative coverage of the surrounding soil with crops. The coverage is expressed through the percentage of groups of crops that share similar characteristics in response to Syrphids.

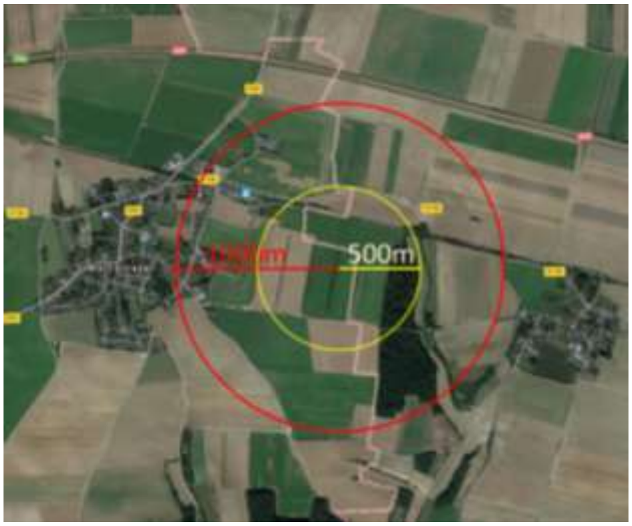


Figure 2: Area with 1000 m and 500 m radius around a sampling point for landscape and crop characterisation

We obtained climatic data from the French national meteorological station located in Boigneville. For the period from 1.1.2008 to 31.12.2011, data about maximum, minimum, average temperature and cumulative rainfall have been collected at daily bases.

2.3 Data preprocessing

Taxonomic and functional diversity of caught syrphid species was described by the total number of species, referred to as species richness and evenness (indicating how abundance is distributed among the species). Indices that combine species richness and evenness into a single value are referred to as diversity indices.

Among the very large number of diversity indices, we decided to calculate the Hill diversity numbers (N0, N1, N2) because they are the easiest to interpret ecologically [4]. The three Hill numbers coincide with the three most important and known measures of diversity: S-number of species, H'-Shannon's index and λ-Simpson's index [3].

Shannon's index H' is calculated as: $H' = -\sum_{i=1}^{S}\left[\left(\frac{n_i}{n}\right)\ln\left(\frac{n_i}{n}\right)\right]$ (S: number of species in the sample, n_i: number of individuals of *i*-th species in the sample, n: number of all individuals in the sample). H' is 0 if there is only one species and its value increases then both the number of species and their evenness increase. For a given number of species, the value of a Shannon diversity index is maximized when all types are equally abundant.

Simpson's index is calculated as: $\lambda = \sum_{i=1}^{S} p_i^2$ (p_i: proportional abundance of the i-th species $p_i = n_i/N$). Simpson's index varies from 0 to 1 and if the community consists of only one species, Simpson's index is 1 and there is no diversity.

The Hill numbers are given in units that represent the effective number, i.e., the number of species in a sample, where each species is weighted by its abundance (N0=>N1=>N2):

- N0 is the number of all species in a sample *i*: N0 = n_i;

- N1 is the number of abundant species and calculated from the Shannon's diversity index H': N1=$e^{H'}$;

- N2 is the number of very abundant species and is based on Simpson's index λ: N2=1⁄λ.

The distribution of abundance among the species is estimated by the evenness index. It is a ratio of observed to maximum diversity and it riches the highest values when the individuals are evenly distributed among species and it is independent of the number of species in the sample. Evenness is calculated as modified Hill number E5:

$$E5 = \frac{\left(\frac{1}{\lambda}\right) - 1}{e^{H'} - 1} = \frac{N2 - 1}{N1 - 1}$$

In addition, we calculated several measures that describe the landscape diversity. Among the most popular metrics used to quantify the landscape composition are the Shannon's index, which emphasizes the richness component of diversity, and Simpson's index, which emphasizes the evenness component [5]. The Shannon's index is therefore recommended for landscape management within an ecological framework. Simpson's index is more responsive to the dominant cover type and is used for specific situations where one cover type is prevailing. These diversity indices can be used to evaluate: i) Landscape richness, which is simply the number of land cover types present within the landscape; ii) Landscape diversity, which evaluates both richness and evenness aspects of the landscape; iii) Landscape evenness, which normalizes for the effect of richness on the diversity index.

Thus, as with the biodiversity metrics, Hill numbers and Evenness index were used, where the number of species was replaced with the number of crops, while the number of individuals of each species with the land cover area (m^2).

For each field the following soil properties have been described: soil texture class and content of clay, sand, coarse fragments, available water holding capacity, and bulk density. The data were acquired from dedicated soil database, established and maintained by ARVALIS.

Phenological development of syrphids (e.g., egg, larval, pupal, and total development times), crops (e.g., leaf unfolding, grass growing, flowering of plants, etc.) and natural vegetation depend on temperature conditions, which are described by degree-days.

Degree-days were calculated using simple logistic equation [1]:

$$DD = \frac{2 * \left[\beta_1 t - \frac{\beta_1 \ln(e^{\beta_3 t + \beta_2} + 1)}{\beta_3}\right]_0^{12}}{24} - MTT$$

According to the study of [2], the minimum daily average temperature of syrphid species is 6^{0}C. For reliability reasons, we selected 5^{0}C for the minimum temperature threshold in our study. Introduction of degree days enables objective comparisons of abundance and diversity dynamic between years and locations, because climatic and environmental conditions for the same calendar date vary between years, therefore calendar days cannot be used as a temporal reference point.

3. RESULTS

With regards to biological control, the abundance of predatory individuals (predators and parasitoids) might be far more relevant for performing pest control than their diversity. In particular, the results of the abundance of the syrphid species in the studied agricultural landscape of Boigneville, show that four syrphid species significantly dominated over other caught species. Comparisons between years show mostly no differences of the top most abundant species nor their relative abundance at a yearly level (Table 1).

Table 1: Relative rank abundance of syrphid species on annual level (2009-2011)

Syrphid species	2009 (%)	2010 (%)	2011 (%)	2009-2011 (%)
Sphaerophoria scripta	43.2	72.0	53.7	55.0
Episyrphus balteatus	22.6	3.6	4.4	10.5
Melanostoma mellinum	9.0	11.1	16.3	12.4
Eupeodes corollae	8.2	3.6	4.4	10.4
Other species	17.0	9.7	8.5	11.8

The prevailing dominance of these four syrphid species has been confirmed also at three week time period (Figure 3). This indicates very low variability of syrhid species at both inter and intra annual level, which further indicates stability of their living conditions, which means that landscape structure and applied crop management have not changed much in the studied period (2009-2011). Abundance is a strong indicator about the syrphid species, but it does not enable investigation of the correlation between taxonomic and functional diversity and landscape elements and crops.

The landscape structure of the studied area (radius of 500 and 1000 m) is very diverse and its diversity does not change over the studied period very much. Table 2 shows landscape diversity and evenness indices for a radius of 500 m around the traps.

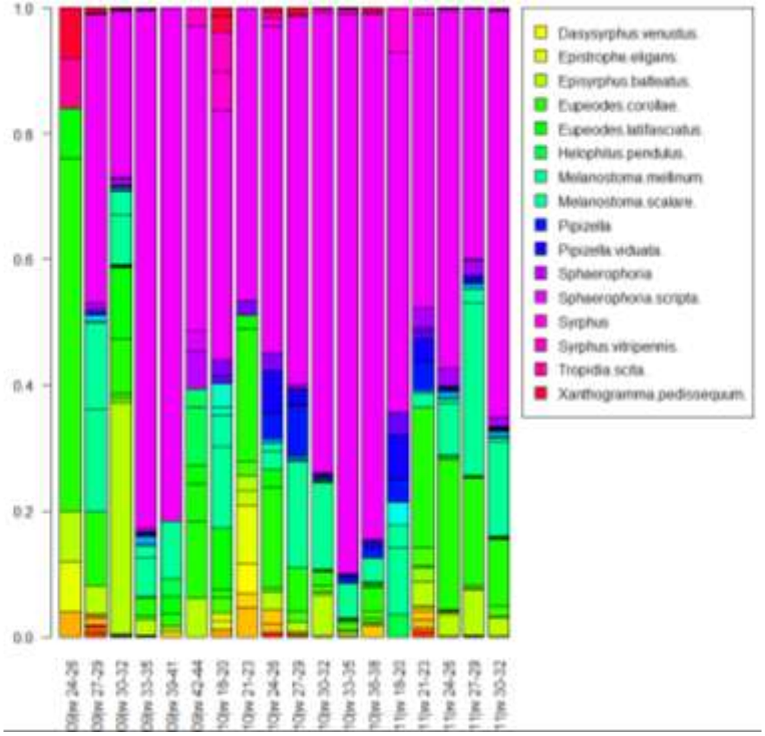


Figure 3: Relative rank abundance of syrphid species for years 2009-2011 at three-week time steps (e.g., 09|w24-26 stands for week 24 to week 26 in year 2009)

Table 2: Diversity of crops in the area within 500 m of the traps

Diversity indices	*Year*	
Number of crops (NO)	2009	15.3
	2010	14.0
	2011	13.2
Number of dominant crops (N1)	2009	7.5
	2010	8.4
	2011	7.2
Number of very abundant crops (N2)	2009	5.5
	2010	6.5
	2011	5.5
Evenness (E5)	2009	0.7
	2010	0.7
	2011	0.7

The prevailing number of crops is high and the value of evenness is relatively high as well. This indicates an even distribution of crops in the area. However, the prevailing crops are cereals, where winter wheat covers the largest areas.

In addition to habitat variability, taxomomic (Table 3) and functional (Table 4) diversity of syrphid species appear to be high and relatively stable at the inter-annual level. Hill number 2 (N2) for taxonomic diversity shows that only the three species are very abundant, therefore the evenness values are rather low.

Table 3: Taxonomic diversity of syrphid species

Year	*Abundance*	*N0*	*N1*	*N2*	*Evenness*
2009	4844	57	4.89	3.18	0.560
2010	3748	48	3.68	2.61	0.599
2011	5469	56	4.26	2.97	0.604
Total	14061	84	4.56	2.98	0.558

Table 4: Functional diversity of syrphid species

Functional groups	*N0*	*N1*	*N2*	*Evenness*
Larvae: terrestrial	31	4.02	2.68	0.514
Larvae: herbal layer	19	3.53	2.49	0.552
Larvae: root zone	14	2.88	2.01	0.493
Overwinter hibernation (OH)	32	4.04	2.68	0.512
OH: above ground surface	9	2.61	1.97	0.610
OH: ground surface	22	3.12	2.25	0.555
OH: root zone	11	5.68	3.90	0.575
Larval food: living plants	6	1.98	1.60	0.558
Larval food: living animals	25	3.72	2.60	0.542
Adult food: nectar flowers	32	4.04	2.68	0.513
Adult food: pollen flowers	12	4.04	2.68	0.573

The analysis of functional diversity of syrphid species shows that larvae of 62% of caught species live in herbal layers and larvae of 81% of species feed on living animals. This indicates very high potential of syrphid species to perform biological pest control because most of syrphid species are feeding on living Aphids (aphidophagous), which are the major pest of cereal crops. The majority (68%) hibernate on the ground, while all adults feed on both nectar and pollen. Such obligatory dependence on nectar and pollen food indicates that the landscape and crop structure provides these required food sources.

Finally, after all the preprocessing of taxonomic, functional and environmental data, we got a dataset comprising seven groups of attributes describing: properties of the fields with sampling traps, taxonomic and functional descriptions of caught syrphid species, soil properties, descriptions of the landscape and crop properties, meteorological conditions with degree-days and descriptions of temporal components of the data collected. The final set of attributes contains in total 209 attributes (Table 5).

4. CONCLUSIONS

Preprocessing of data is a very important step in ecological modelling in general and data mining in particular, because the quality of input data affects the structure of the models and the quality of their predictions. In our study, we used all standard steps to ensure high quality of data, such as data cleaning, outlier detection, missing value treatment, etc.

Table 5: Groups and number of attributes in the final dataset

Group of attributes	*Number of attributes*
Field description	13
Species description	7
Soil description	7
Landscape description	48
Meteorological conditions	7
Temporal component	4
Taxonomical aspect of species	84
Functional aspect of species	39

The majority of our work was focused on transformation and creation of new attributes in order to facilitate the knowledge discovery process about the potential contribution of syrphid species to biological control of Aphid pest species. In order to do this, we used extensive amounts of ecological knowledge about the description of taxonomic and functional properties of the study group of auxiliary species. The completeness and quality of the obtained (preprocessed) data were reviewed and confirmed by ecological experts.

We conclude that landscape and crop diversity support high taxonomic and functional diversity of syrphids. This is a very promising preliminary approximation, which indicates that we can expect to obtain interesting results from data mining models. Therefore, the next step is to apply various data mining methods on the preprocessed dataset in order to discover new knowledge about interactions between the environment (landscape structure, crop management, soil, climate) and taxonomic and functional diversity of syrphid species. The new knowledge will be used for enhancing the existing syrphid species to perform efficient biological pest control on growing crops.

5. REFERENCES

[1] Caicedo, D.R., Torres, J.M.C., Cure, J.R. 2012. Comparison of eight degree-days estimation methods in four agroecological regions in Colombia. *Agrometeorology* 71, 299-307.

[2] Hassall, C., Owen, J., Gilbert, F. 2017. Phenological shifts in hoverflies (Diptera: Syrphidae): linking measurement and mechanism. *Ecography* 40, 853–863.

[3] Legendre P., Legendre L. 2012. Numerical ecology. Elsevier, Amsterdam, Netherlands.

[4] Ludwig, J.A., Reynolds, J.F. 1988. Statistical Ecology. New York, Chichester, Brisbane, Toronto, Singapore, John Wiley & Sons.

[5] Querner, P., Bruckner, A., Drapela, T., Moser, D., Zaller, J.G., Frank, T. 2013. Landscape and site effects on Collembola diversity and abundance in winter oilseed rape fields in eastern Austria. *Agr Ecosyst Environ* 164, 145–154.

[6] Speight, M.C.D., Castella, E. 2016. StN Content and Glossary of terms. Syrph the Net, the database of European Syrphidae (Diptera), Vol. 94, 89 pp, Syrph the Net publications, Dublin.

[7] Witten, I.H., Frank, E. 2011. Data Mining: Practical Machine Learning Tools and Techniques - 3rd edition. Morgan Kaufmann.

A Comparison of DEXi, AHP and ANP Decision Models for Evaluation of Tourist Farms

Tanja Dergan
Institut Jožef Stefan
Jamova cesta 39, 1000 Ljubljana
Slovenia
tanja.dergan@ijs.si

ABSTRACT

This research covers a comparison between decision models created with the DEXi tool based on the DEX methodology, and the decision models made by Super Decision tool using Analytical hierarchical process (AHP) and Analytical network process (ANP) methodology, for analysing decisions, on a case study of tourist farms. Based on the performance of decision models, the most appropriate decision making methodology that maximally satisfied the evaluation criteria was DEX. Based on the empirical data, the best evaluated farm was Tourist farm IV from Logarska dolina, which achieved the best evaluation by all three decision models.

Keywords:

Decision support modelling, DEXi, Analytical hierarchical process, Analytical network process, touristic farm

1. INTRODUCTION

Decision making can be defined as a cognitive mind process, a human quality used to solve everyday situations. Some decisions may be felt as easy, others difficult and demanding. The development of the decision support system represents a new step towards optimisation and improvement of the whole decision making process [1].

Multi-criteria decision making is an approach where we make decisions on the basis of several criteria. This approach is necessary when intuitive decision making is not sufficient, either because of conflicts between criteria, or because of the differences between decision makers.

Tourist farms provide an important development potential for the inhabitants of the Slovenian countryside [2]. There are currently over 800 tourist farms in Slovenia and their number is growing rapidly. The increasing number of tourist farms brings many benefits for the regional development and local inhabitants such as prevention of emigration of young people from the countryside, preservation of the cultural landscape and the provision of social security for farming families with an additional source of income.

The problem that we deal with, relates to the selection of an appropriate modelling method for the case study of tourist farms and the criteria that they should fulfil in order to be chosen by users (tourists), for spending their holidays.

The goal of this article is to assess the decision models built with three different decision making methodologies, which is based on comparisons of their complexity, interdependency and consistency. The objective of these decision making models was also to help potential consumers to make decisions about which tourist farm to choose.

2. DATA DESCRIPTION

The study evaluates four tourist farms, whose locations are in different geographical regions of Slovenia: I - Izola, II - Pohorje, III - Ponikva, IV - Logarska dolina. In addition to the location of the selected tourist farms, it was also important that they provide the possibility of overnight stays. At this stage, 6 criteria and 17 sub-criteria were selected, (Table 1), and used in all decision models. The criteria were chosen personally bay the researcher, while the data for the evaluation of tourist farms were obtained through personal interviews of farm owners and guests, as well as survey questionnaires of potential guests. The data obtained through the interviews of the farm owners and guests have given numerical values, while the survey questionnaires of potential guest resulted in descriptive data.

Criteria	**Location**	**Form of supplementary activity**	**Offer: complementary activity**
Sub-criteria	-Where is it located -Accessibility	-Stationary farm -Excursion farm	-Food -Drink -Sport activities -Living space -Tourist farm Logo
Criteria	**The surrounding of the farm**	**Hospitality**	**Host to guest relationship**
Sub-criteria	-Flowers and greenery -Preserving the cultural landscape -Categorization	-Reception -Events -Access to information for guests	-Family arrangement -Cleanliness

Table 1: Structure of data.

3. DECISION MODELS

Different methods such as Servqual and Dematal [5] can be used to measure the quality of tourist services. Some of them also enable work with inaccurate and incomplete data and use an interval account such as Mund, Promethee [5]. However, in this study we focused on the decision methods for construction of DEXi, AHP and ANP decision models.

3.1 DEXi

DEX is a multi-attribute methodology for decision making. The methodology is based on attributes with a finite set of qualitative values instead of attributes with numerical values [8]. The DEX methodology enables a construction of transparent and comprehensive models and it provides techniques for integration of attributes through aggregation rules in form of hierarchical decision trees.

DEXi is a software modelling tool, which is based on the DEX methodology and facilitates the development of qualitative Multi Attribute Decision Models (MADM) and enables an evaluation and what-if analysis of decision options [8]. DEXi is useful in cases where we do not have numerical data or ratings, but only qualitative ones [3]. In general, DEXi models are customised and do not have a complex structure, are insensitive to minor changes in input data and capable of resetting procedures [6].

In the DEXi modelling tool, the alternatives are described by initial attributes, which are then evaluated separately according to their values. The final evaluation of the alternatives is obtained by an aggregation process of input data (values of initial attributes Xi) using aggregation functions Fi. The output value of the topmost node in the decision tree (decision model) is used for selection of the most suitable alternative among all evaluated alternatives.

The DEXi model was applied to evaluate four tourist farms using data derived from interviews, as well as survey questionnaires. The tourist farms and their regulatory standards were precisely defined, in order to select attributes, which have been structured in the DEXi model. The goal of the model was to decompose the problem into smaller sub-problems, which were assessed individually using criteria determined by the decision maker: For example, the set of values for the attribute "Sport Activities" was: excellent; medium and poor. Through the process of hierarchical integration, using the utility functions obtained provided by the decision maker, the final assessment of the top-most attribute was determined. The outcome of the evaluation was an assessment of tourist farms.

3.2 Analytical hierarchical process (AHP)

AHP is an established and well-researched method of analysing a hierarchical decision-making processes based on mathematics and psychology [7]. The model was built in the Super Decision modelling tool [4] and consists of a general goal (selection of the best tourist farm), criteria and sub-criteria (Table1) and common options or alternatives (Tourist farm I, Tourist farm II, Tourist farm III and Tourist farm IV). The structure enables the possibility for taking into account the given elements at the selected level as well as all elements at lower levels. The criteria and their hierarchical structure are the same in the DEXi models as well as in AHP models, which provides the basis for comparisons of the models. The tourist farm is an alternative and therefore lies the highest in the hierarchy tree. They are determined by subordinate criteria, and further by lower sub-criteria. The method converts the evaluation of tourist farms into numerical values that can be processed and compared for each criteria in the hierarchy. Mutual pairwise comparisons of alternatives (tourist farms) were performed in a hierarchical model based on the obtained data (surveys, interviews). A basic scale (i.e., the Saaty scale) from 1 to 9 was used, where each gives a specific preference [7]. The use of numerical weights allows for rational and consistent comparison of different or incompatible elements with each other. The results in the AHP method are interpreted in three ways: i) 'Normals', where the results are presented in the form of priorities, where each one of the alternatives are summed and then each element is divided by the sum, ii) 'Ideals', where the values are obtained from 'Normals' by dividing each of its entries by the largest value in the column, so that the best alternative gets a priority of 1 and the others get proportion less than 1, and iii) 'Raw', whose values are read directly from the Limit Supermatrix.

3.3 Analytical Network Process (ANP)

The ANP is implemented in the software Super Decisions and has been applied to various problems both to deal with decisions and to illustrate the uses of the new theory. The ANP is a coupling of two parts. The first consists of a control hierarchy or network of criteria and subcriteria that control the interactions in the system under study. The second is a network of influences among the elements and clusters [7]. In the Super Decision modelling tool, the criteria were grouped into a network model, with clusters and with related criteria, and not in a hierarchical level. The method allows for interactions and feedbacks within the cluster, as well as between clusters, for example the alternatives of the decision in another cluster. This helps to make the basic computer operations and logical multiplication in different ranges as required by the model. The mutual pairwise comparisons were performed like in AHP model, base on the Saaty scale [7] from 1 to 9 meaning: 1) criteria are the same, 2) criteria is equivalent to another or has a moderate advantage over another, 3) criteria has a moderate advantage over other criteria, 4) criteria has a moderate to great advantage over another criteria, 5) criteria has a great advantage over another criteria, 6) criteria has a great advantage over other criteria, 7) criteria has a very great advantage over another criteria, 8) criteria has a very large to an extremely high advantage over another criteria and 9) criteria has an extremely high advantage over another criteria. Our research focuses on development of three ANP model applications: i) the simplest single model, which was built only in one layer and was the easiest to build, ii) the two-layers model, which divide the model into upper and lower levels, and iii) the complex three-layers model, which consists of several layers of sub-networks and is one of the most demanding models in the presented research. The input data, as well as the data from interviews and surveys used in the ANP model was excerpt from the previously presented AHP and DEXi models. However, due to the complexity of the ANP complex three-layers model, the criteria are further subdivided to create a more extensive model.

4. RESULTS WITH DISCUSSION

The aim of the research was to compare DEXi, AHP and ANP multi-criteria models, for an evaluation of tourist farms.

4.1 DEXi model

In DEXi modelling, the main focus is on the rationality and the regularity of the criteria. Based on the obtained data, a multi-criteria model was developed that maximally met the given criteria. The best evaluated tourist farm (Figure 1) is from Logarska dolina (Tourist farm IV). The second best tourist farm was Tourist farm II, which achieved the same level of evaluation scores in almost all criteria as Tourist farm IV. Tourist farm II was evaluated worse only in the criterion "Complementary activities" (Figure 1). Tourist farm IV and Tourist farm II got the highest score in the criteria "The surrounding of the farm". However, in the criteria "Tourist Farm Logo" and "Where is it located" they did not receive the best estimates. Tourist farm III was evaluated well and it is potentially a good choice for tourists. The worst evaluated was Tourist farm I, although it was very well evaluated for the criteria “The surrounding of the farm”. The Tourist farm I was inadequate due to poor estimates of the criteria “Complementary activities” and “Hospitality” (Figure 2).

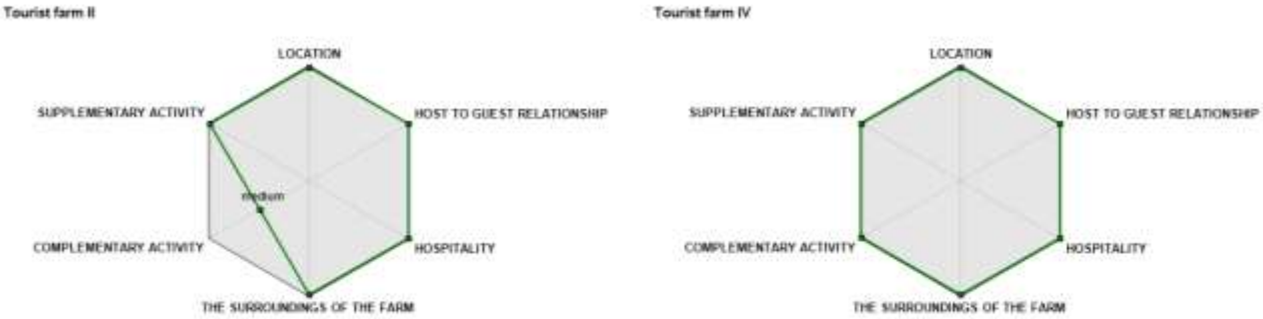


Figure 1: Evaluated criteria in DEXi model for Tourist farm II and Tourist farm IV.

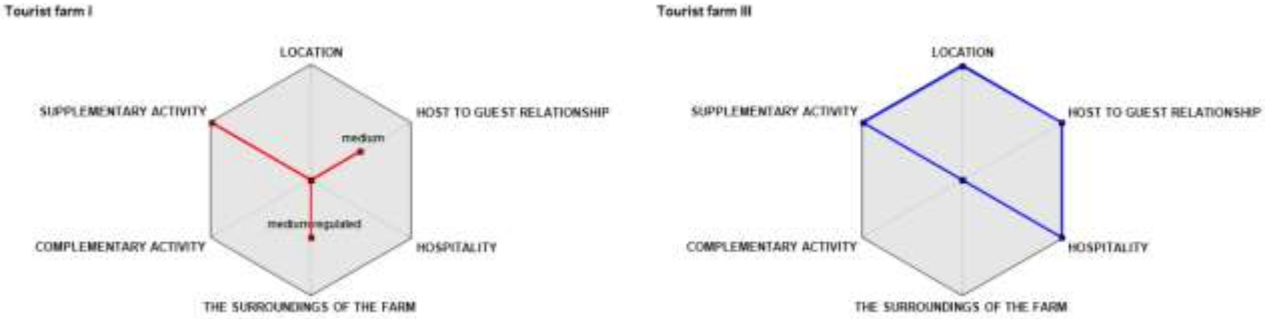


Figure 2: Evaluated criteria in DEXi model for Tourist farm I and Tourist farm III.

4.2 AHP model

In AHP method, the development of decision-making model (identification of the decision problem, identification of the alternatives and determination of criteria) was similar as in DEXi. The difference in AHP is in the pairwise comparison of the criteria with respect to the goal and the pairwise comparison of the alternatives. The results of the AHP model (Figure 3), based on the Normals values show that the best evaluated tourist farm was from Logarska dolina (Tourist farm IV) which received 40%. Tourist farm II received 37%, Tourist farm I received 13 % and Tourist farm III received the lowest percentages 10%. Based on Ideals values (Figure 3), the results can be interpreted also in the way as: Tourist farm I is 33% as good as a Tourist farm IV, Tourist farm II is 92% as good as Tourist farm IV and Tourist farm III is 25% as good as Tourist farm IV.

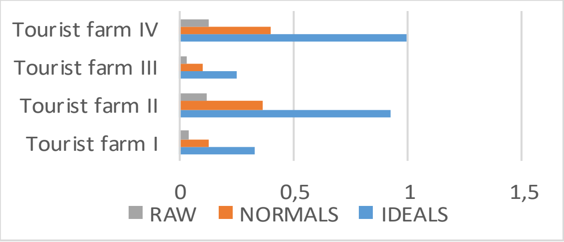


Figure 3: Synthesized priorities for the alternatives in AHP model.

The criteria “Complementary activity” and “Location” contributed the highest values, while the least impact on the final results had criteria “Form of supplementary activity”.

4.3 ANP model

Three different models have been developed with the ANP method in the Super Decision tool. According to the input data and the problems that were considered, we came to the conclusion (Table 2) that the outputs for all three applications of the ANP methods show that the Tourist farm IV from Logarska dolina was selected as the most appropriate one. The tourist farm IV received the highest percentage in the three layered model (40%), in the two layered model 38% and in single layered model 36%. The lowest percentage achieved Tourist farm III in the three layered model (10%) and 17% in both two layered and single layered model. The results in the simplest single layered model, showed that the greatest impact on the final results had criteria “Complementary activity” and “The surrounding of the farm”, while the criteria “Location” heed the least influence on the results. In the two-layereds model, Tourist farm IV got the best assessment in all criteria. Tourist farm II received good assessment of the criteria “Host to guest relationship”, “Form of supplementary activity”, “Hospitality” and “Complementary activity”. Tourist farm I received good assessment for the criteria “Location2 and “Form of supplementary activity”. Tourist farm III was assessed with lower estimates, only a slightly higher rating according to the criteria “The surrounding of the farm”. In the complex three-layereds model, Tourist farm IV achieved the best assessment in the criteria “Costs” and “Priority”. Among all the criteria, three were selected that were assessed as most important and that had the greatest impact on the final results: “Where is it located”, “Stationary farm” and “Cleanliness”. By comparing results of all three ANP model applications, a deviation in the percentages was observed. The percentages of the three layered model were for Tourist farm IV and Tourist farm II evaluated higher and for Tourist farm III and Tourist farm I the percentages were evaluated lower in comparison with the percentages of the single and two layered models, where only a smaller deviation of the percentage occurred (Table 2). The three layered model contained in the upper level two new clusters, Strategic and Basic. Strategic served in the model for evaluating the Basic criteria using the rating model. The basic cluster consists of additional criteria: “Priority”, “Cost” and “Risk”. The new criteria contain lower levels, where the already known criteria and sub-criteria used in DEXi and AHP models are added (Table1). The criteria were grouped into i) Priority criteria: “Location” and “Hospitality”, ii) Cost criteria: “Complementary activity” and “Form of supplementary activity” and iii) Risk criteria: “Host to guest relationship” and “The surrounding of the farm”. The presented expansion of the three-layered model led to a deviation of the percentage (Table 2).

Table 2: ANP total results.

	ANP-Single layered model	**ANP- Two layered model**	**ANP-Three layered model**
Tourist farm I	21%	20%	15%
Tourist farm II	26%	25%	35%
Tourist farm III	17%	17%	10%
Tourist farm IV	36%	38%	40%

5. CONCLUSIONS

The goal of this study was comparison of the DEXi tool in the DEX methodology, and the Super Decision tool in the AHP and ANP methodology. The study presented three examples of decision models for evaluation of tourist farms, which in addition to the final results, also provided appropriate measures to improve the offers of the more poorly assessed alternatives. The objective of these decision making models used in the survey was also to help potential consumers to make decisions about the most appropriate tourist farms. The results of all models indicated that the Tourist farm IV located in Logarska dolina received the best evaluation results and represents the best provider of touristic activities in the farm. The applied methodologies were found to be very successful and effective. DEXi tool has proved a very simple model, which is not related to numerical values of input data. It is also easy to add additional criteria to a structured decision tree, despite the measurements of input data and integration functions. The modelling software indicates where model needs to be modified or corrected due to an additionally added criterion. AHP in the Super Decision tool also presents a simple and transparent way to build a model and represents a very good alternative to the DEXi model. However, in contrast to DEXi, it is bound to numerical values. The ANP model build in the Super Decision tool proved to be more complicated. The application of the simple single network model and the two-layers model are comprehensive, and still sufficiently understandable, while the approach to build a complex three-layers model is much more complex. Despite the fact that, due to its size, the three-layers model gives us more precise results, its use is more difficult.

This led to the conclusion that in order to create a model with qualitative attributes, perform what-if analysis and include additional alternatives for evaluation (e.g., Tourist farm 5, Tourist farm 6., etc.), the most appropriate model was built in DEXi. The AHP model, which is based on numerical values of input data, can be more precise then the DEXi model, but on the other hand, determining one value in the comparison matrix in AHP, is much more difficult then determining it in DEXi models. Thus, if we want to use quantitative attributes without the evaluation of additional alternatives, the AHP methods could be applied. The application of the three-layers ANP model is non-transparent due to multiple layers and its construction is highly time-consuming. The use of a single or two-layer ANP model in the Super Decision toll can give sufficiently more precise results.

Each of the assessed decision modelling methods has its advantages and disadvantages. However, we have found that DEXi is the best modelling approach for the assessment of tourist farms, and that the Tourist farm IV located in Logarska dolina represents the best provider, which was not confirmed only by the DEXi model, but also by the models developed in Super Decision tool with AHP and ANP method.

6. REFERENCES

[1] Babčec R. September 2010. Analysis functionality and system performance to support group decision-making thesis. Faculty of organizational sciences, UM. +

[2] Deklava M. 1998. Development of tourism places. Participation of the villagers. Ljubljana, Tourist Association

[3] Dergan T. 2010. Development of Multi-criteria model for tourism farm assessment. Graduate thesis. UM. +

[4] Dergan T. 2014. Development and application of AHP and ANP decision models for farm tourism analysis. Master’s thesis. University Maribor

[5] Jereb E, Bohanec M, Rajkovič V. 2003. DEXi computer program for multi-parameter decision making. Kranj, Modern organization.

[6] Murn T. 2005. Learning for the Future Multi-Attribute Decision making Models and the sistem DEXi for schools. Karlstad, City Tyrick: 329-343 str.

[7] Saaty R.W. 2003. Decision making in complex environments – Analytical hierarchical process (AHP) for Decision Making and Analytical network process (ANP) for Decision Making with Dependence and Feedback. University of Pittsburgh.

[8] Zadnik S.L, Žerovnik J, Kljajić B.M, Drobne S. 2015. Proccedings Sor 15. The 13th International Symposium on operational research. Bled. Slovenia

A State-Transition Decision Support Model for Medication Change of Parkinson's Disease Patients

Biljana Mileva Boshkoska[1,2]
biljana.mileva@ijs.si

Dragana Miljković[1]
dragana.miljkovic@ijs.si

Anita Valmarska[1,4]
anita.valmarska@ijs.si

Dimitris Gatsios[3]
dgatsios@cc.uoi.gr

George Rigas[3]
george.a.rigas@gmail.com

Spiros Konitsiotis[3,5]
skonitso@gmail.com

Kostas M. Tsiouris[3]
tsiourisk@biomed.ntua.gr

Dimitrios Fotiadis[3]
fotiadis@cc.uoi.gr

Marko Bohanec[1]
marko.bohanec@ijs.si

ABSTRACT

In this paper, we present a state-transition decision support model for medication change of patients with Parkinson's disease (PD), implemented with method DEX. Today, PD patients can be treated with three basic medications: levodopa, dopamine agonist, MAO-B inhibitors, and their combinations. We propose a model which, based on the current patient's symptoms (motor symptoms, mental problems, epidemiologic data and comorbidities), suggests how to change the medication treatment given the patient's current state. The model is based on expert's knowledge of neurologists and is composed of (1) a state-transition model that presents all possible medication changes, and (2) decision rules for triggering the changes, represented in terms of a qualitative rule-based multi-criteria model. The model assesses all states described by the state-transition model and proposes multiple different yet still correct possibilities for medication change.

Categories and Subject Descriptors

H.4.2 [Types of Systems]: Decision support.
J.3 [Life and Medical Sciences]: Medical information systems.

General Terms

Algorithms, Management, Measurement, Design, Human Factors

Keywords

Parkinson's disease, medication change, decision model

1. INTRODUCTION

Parkinson's disease (PD) is a complicated, individual degenerative disorder of the central nervous system for which there is no cure. Hence it requires a long-term, interdisciplinary disease management including typical medicament treatment with levodopa (LD), dopamine agonist (DA), and enzymes (E), such as MAO-B inhibitor. Due to the different combinations of motor and mental symptoms from which PD patients suffer, in addition to existing comorbidities, the interchange of medications and their combinations is patient-specific [1]. In the framework of the EU Horizon 2020 project PD_manager (http://www.parkinson-manager.eu/) [2] we developed a decision support model, called the "How" model, for PD management which suggests how to change the medication treatment given patients' current state. The assessment is based on data that include patients' motor symptoms (dyskinesia intensity, dyskinesia duration, offs duration), mental problems (impulsivity, cognition, hallucinations and paranoia), epidemiologic data (patient's age) and comorbidities (cardiovascular problems, hypertension and low blood pressure). The model is composed of (1) a state-transition model that presents the medication change among levodopa, dopamine agonist, MAO-B inhibitors and their combinations, and (2) decision rules for triggering the changes, represented in terms of a qualitative multi-criteria model. The latter has been developed using the DEX method [4], which integrates the qualitative multi-criteria decision modeling with rule-based expert systems.

2. MODEL DESIGN

The model development was performed with neurologists who work with PD patients. The process of decision analysis led to a design of a model composed of two key elements: (1) A state-transition model that represents all possible combinations of used medicaments and transitions between them, and (2) a multi-criteria DEX model that provides decision rules for each transition.

2.1 A state-transition model

In the state-transition model the medication treatments (states) and transitions among them are represented in a form of a cube as presented in Figure 1. In Figure 1, each medication-treatment state is as a circle and each change of medication treatment is represented with a directed arc. Each state corresponds to the set of medications that constitute the current treatment. The set can be empty (the symbol O indicates no medication therapy), or can consist of any combination of DA, LD and E (Enzymes, such as MAO-B inhibitor). For example, the state DA+E means that the current medication treatment of the patient consists of dopamine agonist (DA) and MAO-B inhibitor. From this state there are three possible state changes depending on the combinations of patient's symptoms: add LD to the treatment (state denoted as LD+DA+E), remove DA from the current treatment (state E) or remove E and use only DA (state DA).

The absence of a directed arc between two states means that a particular change of medication treatment is not addressed in the model, either because it has been deliberately excluded (transitions from and to state O, which are out of scope of the PD_manager project), or is rarely or not at all used in practice. A

[1]Jožef Stefan Institute, Ljubljana, Slovenia
[2]Faculty of information studies in Novo mesto, Slovenia
[3]Unit of Medical Technology and Intelligent Information Systems, University of Ioannina, Greece
[4]Jožef Stefan International Postgraduate School, Ljubljana, Slovenia
[5]Department of Neurology, Medical School, University of Ioannina, Ioannina, Greece

reflexive arc means an increase/decrease of the medication (dosage or intake) [3].

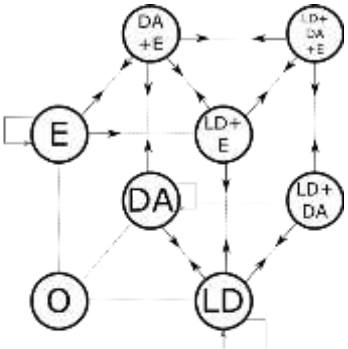


Figure 1: A state-transition model for medication change among levodopa (L), dopamine agonist (DA), MAO-B inhibitors (E) and their combinations. Symbol O represents the state where the patient does not take medications.

2.2 DEX model

The transitions in the state-transition model (Figure 1) are triggered according to a multi-attribute model, which is responsible for interpreting patients' motor symptoms, mental problems, epidemiologic data and comorbidities, and aggregating them into an overall assessment of the potential medication changes of a given patient. The model is hierarchical and qualitative, developed using a qualitative multi-attribute modelling method DEX [4]. DEX models decompose the decision problem into smaller, less complex sub problems, which are represented by a hierarchy of attributes. Attributes from the decision alternatives are aggregated in order to obtain an overall the evaluation or recommendation. DEX belongs to the class of qualitative multi-criteria decision making methods: it uses qualitative (discrete) variables instead of quantitative (numerical) ones, and employs decision rules rather than numerical aggregation functions for the aggregation of attributes. The method DEX is supported by DEXi [5], freely available software that supports both the development of DEX models and their application for the evaluation and analysis of decision alternatives. DEX was chosen for modelling due to its previous successful usage for implementation of decision support models in health care [6][7].

Using DEX principles of model development, the state-transtion model from Figure 1 is mapped into a qualitative multi-attribute model presented in Figure 2. The model consists of basic and aggregated attributes given in a structure that identifies possible transitions in the state-transition diagram for a given patient [8]. The model combines 22 basic attributes including data about motor symptoms (rigidity, tremor, and bradykinesia), mental problems (impulsivity, cognition, hallucinations, paranoia), comorbidities (cardiovascular, low blood pressure, hypertension), and dyskinesia (offs duration, intensity, and duration). In addition, there is data about patient's age and activity, and data about the current therapy (which medications is the patient currently using, and whether or not the maximum dosages of DA and LD have been reached). The values of these attributes constitute model's inputs.

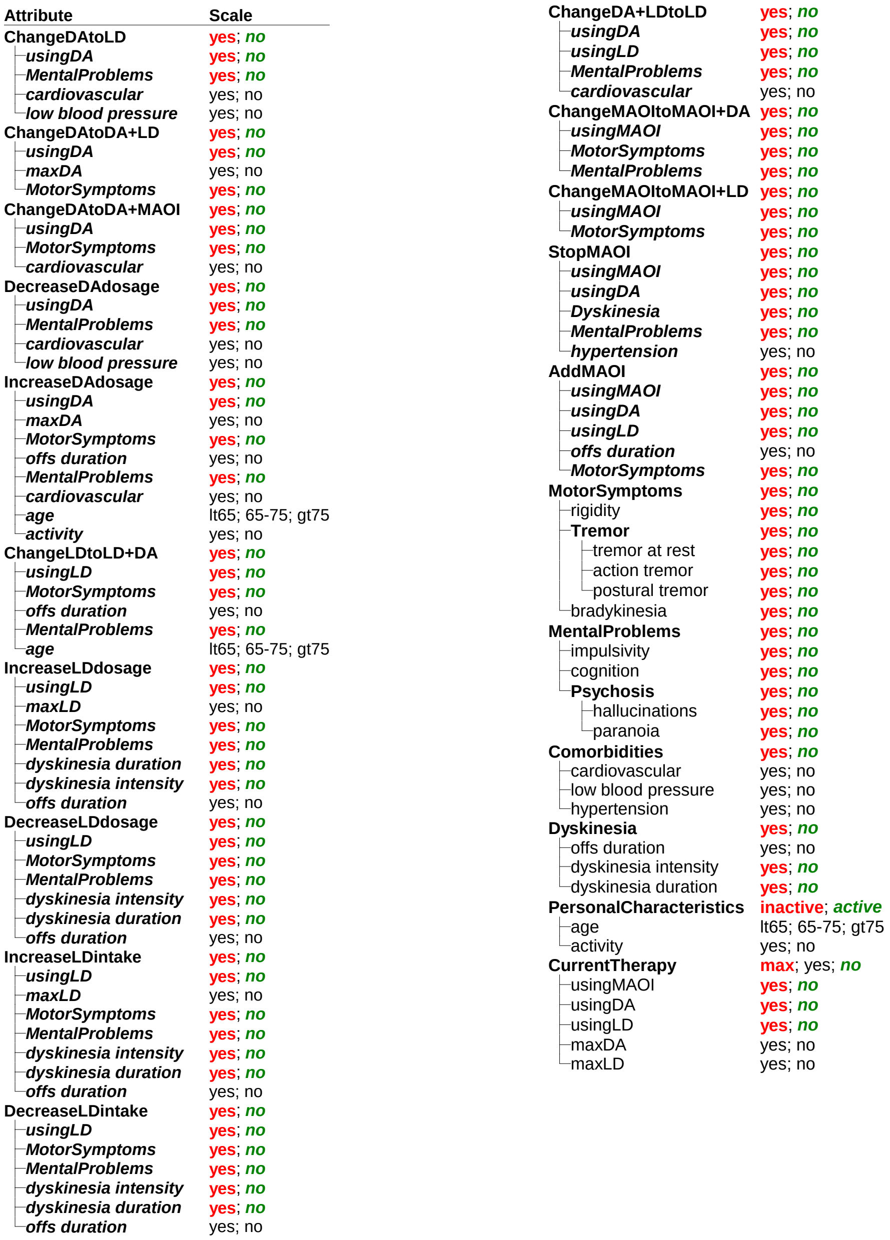
Aggregation of the basic attributes leads to two sets of attributes. The first set is composed of six aggregated attributes: Dyskinesia, MotorSymptoms, CurrentTherapy, PersonalCharacteristics, Comorbidities and MentalProblems. The purpose of this set of attributes is to aggregate several specific symptoms into common indicators, which are used as inputs to the second set of aggregated attributes. For instance, Dyskinesia is a common indicator of patient's involuntary movements caused as a side effect of medications; it is determined by aggregating the basic attributes *offs duration*, d*yskinesia intensity*, and *dyskinesia duration*.

The second group of aggregated attributes forms a set of 15 submodels, which determine the transitions from one medication state to the other one as given in the state-transition diagram (Figure 1). Those submodels are the following:

1. **ChangeDAtoLD:** Change therapy from dopamine agonist to levodopa
2. **ChangeDAtoDA+LD:** Change therapy from dopamine agonist to dopamine agonist and levodopa
3. **ChangeDAtoDA+MAOI:** Change therapy from dopamine agonist to dopamine agonist and MAO-B inhibitors
4. **DecreaseDAdosage:** Decrease the dosage of dopamine agonist
5. **IncreaseDAdosage:** Increase the dosage of dopamine agonist
6. **ChangeLDtoLD+DA:** Change therapy from levodopa to levodopa and dopamine agonist
7. **IncreaseLDdosage:** Increase the dosage of levodopa
8. **IncreaseLDintake:** Increase the intake of levodopa
9. **DecreaseLDintake:** Decrease the intake of levodopa
10. **DecreaseLDdosage:** Decrease the dosage of levodopa
11. **ChangeDA+LDtoLD:** Change therapy from dopamine agonist and levodopa to levodopa
12. **ChangeMAOItoMAOI+DA:** Change therapy from MAO-B inhibitors to MAO-B inhibitors and dopamine agonist
13. **ChangeMAOItoMAOI+LD:** Change therapy from MAO-B inhibitors to MAO-B inhibitors and levodopa
14. **StopMAOI:** Stop using MAO-B inhibitors
15. **AddMAOI:** Add MAO-B inhibitors to the current therapy.

At the top of each submodel, there is the *root* attribute which represents the overall assessment of medication change under consideration. For example, the submodel **ChangeDA+LDtoDA** estimates the change of medication from dopamine agonist and levodopa to dopamine agonist based on the information whether the patient already takes DA (*usingDA*) and LD (*usingLD*), if the patient has increased mental problems (*MentalProblems*) and/or cardiovascular problems (*cardiovascular*).

All submodels were obtained through expert modelling. In this case, decision-support models were developed in collaboration between the neurologists (experts) from and the decision analyst. The work proceeds in the form of a question-answer dialogue, led by the analyst, aimed at identifying the important indicators and decision rules used, implicitly or explicitly, by the expert when making decisions.



Attribute	Scale
ChangeDAtoLD	**yes**; ***no***
usingDA	**yes**; ***no***
MentalProblems	**yes**; ***no***
cardiovascular	yes; no
low blood pressure	yes; no
ChangeDAtoDA+LD	**yes**; ***no***
usingDA	**yes**; ***no***
maxDA	yes; no
MotorSymptoms	**yes**; ***no***
ChangeDAtoDA+MAOI	**yes**; ***no***
usingDA	**yes**; ***no***
MotorSymptoms	**yes**; ***no***
cardiovascular	yes; no
DecreaseDAdosage	**yes**; ***no***
usingDA	**yes**; ***no***
MentalProblems	**yes**; ***no***
cardiovascular	yes; no
low blood pressure	yes; no
IncreaseDAdosage	**yes**; ***no***
usingDA	**yes**; ***no***
maxDA	yes; no
MotorSymptoms	**yes**; ***no***
offs duration	yes; no
MentalProblems	**yes**; ***no***
cardiovascular	yes; no
age	lt65; 65-75; gt75
activity	yes; no
ChangeLDtoLD+DA	**yes**; ***no***
usingLD	**yes**; ***no***
MotorSymptoms	**yes**; ***no***
offs duration	yes; no
MentalProblems	**yes**; ***no***
age	lt65; 65-75; gt75
IncreaseLDdosage	**yes**; ***no***
usingLD	**yes**; ***no***
maxLD	yes; no
MotorSymptoms	**yes**; ***no***
MentalProblems	**yes**; ***no***
dyskinesia duration	**yes**; ***no***
dyskinesia intensity	**yes**; ***no***
offs duration	yes; no
DecreaseLDdosage	**yes**; ***no***
usingLD	**yes**; ***no***
MotorSymptoms	**yes**; ***no***
MentalProblems	**yes**; ***no***
dyskinesia intensity	**yes**; ***no***
dyskinesia duration	**yes**; ***no***
offs duration	yes; no
IncreaseLDintake	**yes**; ***no***
usingLD	**yes**; ***no***
maxLD	yes; no
MotorSymptoms	**yes**; ***no***
MentalProblems	**yes**; ***no***
dyskinesia intensity	**yes**; ***no***
dyskinesia duration	**yes**; ***no***
offs duration	yes; no
DecreaseLDintake	**yes**; ***no***
usingLD	**yes**; ***no***
MotorSymptoms	**yes**; ***no***
MentalProblems	**yes**; ***no***
dyskinesia intensity	**yes**; ***no***
dyskinesia duration	**yes**; ***no***
offs duration	yes; no
ChangeDA+LDtoLD	**yes**; ***no***
usingDA	**yes**; ***no***
usingLD	**yes**; ***no***
MentalProblems	**yes**; ***no***
cardiovascular	yes; no
ChangeMAOItoMAOI+DA	**yes**; ***no***
usingMAOI	**yes**; ***no***
MotorSymptoms	**yes**; ***no***
MentalProblems	**yes**; ***no***
ChangeMAOItoMAOI+LD	**yes**; ***no***
usingMAOI	**yes**; ***no***
MotorSymptoms	**yes**; ***no***
StopMAOI	**yes**; ***no***
usingMAOI	**yes**; ***no***
usingDA	**yes**; ***no***
Dyskinesia	**yes**; ***no***
MentalProblems	**yes**; ***no***
hypertension	yes; no
AddMAOI	**yes**; ***no***
usingMAOI	**yes**; ***no***
usingDA	**yes**; ***no***
usingLD	**yes**; ***no***
offs duration	yes; no
MotorSymptoms	**yes**; ***no***
MotorSymptoms	**yes**; ***no***
rigidity	**yes**; ***no***
Tremor	**yes**; ***no***
tremor at rest	**yes**; ***no***
action tremor	**yes**; ***no***
postural tremor	**yes**; ***no***
bradykinesia	**yes**; ***no***
MentalProblems	**yes**; ***no***
impulsivity	**yes**; ***no***
cognition	**yes**; ***no***
Psychosis	**yes**; ***no***
hallucinations	**yes**; ***no***
paranoia	**yes**; ***no***
Comorbidities	**yes**; ***no***
cardiovascular	yes; no
low blood pressure	yes; no
hypertension	yes; no
Dyskinesia	**yes**; ***no***
offs duration	yes; no
dyskinesia intensity	**yes**; ***no***
dyskinesia duration	**yes**; ***no***
PersonalCharacteristics	**inactive**; ***active***
age	lt65; 65-75; gt75
activity	yes; no
CurrentTherapy	**max**; yes; ***no***
usingMAOI	**yes**; ***no***
usingDA	**yes**; ***no***
usingLD	**yes**; ***no***
maxDA	yes; no
maxLD	yes; no

Figure 2: Structure and value scales of the "How" medication change model

Figure 2 shows the value scales and structure of the model. It shows that most attributes in the model are binary, each taking one of the two corresponding values: *yes* or *no*. Coloured values indicate that the corresponding attribute is ordered from left-to-right, so that the leftmost (**red**) value indicates a problematic, and the rightmost (***green***) a non-problematic patient's condition. The red/left values generally indicate a problem that should be addressed by medication change.

2.3 Decision rules

For each aggregate attribute in the DEX model, it is necessary to define the values of that attribute for all possible combinations of lower-level (input) attribute values. For example, the **IncreaseLDdosage** aggregate attribute depends on seven lower level attributes that correspond to current patients' medication treatment and symptoms. These attributes are binary, so there are 2^7 = 128 $2^7 = 128$ possible combinations of their values. The DEXi software was used to represent, manage and define such combinations in the form of decision tables. All decision rules contained in the model are presented in a tabular form together with a verbal interpretation. Table 1 is an example of a decision table that defines the decision rules for the aggregated attribute **ChangeDatoLD**. The symbol '*' used in the decision tables denotes any value that can appear at that position. For instance, in connection with an attribute than can take the values "yes" and "no", the '*' stands for "yes or no".

According to the decision rules presented in Table 1, one may read that the change of medication treatment from DA to LD should happen only when the patient already takes DA (*usingDA*). The change may take place in in three different cases: the patient has mental problems, cardiovascular problems, or low blood pressure. Otherwise, the change to LD should not happen.

The whole model contains 21 other decision tables such as Table 1, corresponding to the remaining aggregate attributes in the model.

Table 1: Decision rules for submodel ChangeDatoLD

	usingDA	Mental-Problems	cardio-vascular	low blood pressure	Change-DAtoLD
1	**yes**	**yes**	*	*	**yes**
2	**yes**	*	yes	*	**yes**
3	**yes**	*	*	yes	**yes**
4	*	***no***	no	no	***no***
5	***no***	*	*	*	***no***

3. CONCLUSIONS AND FUTURE WORK

Using the DEX method, we developed a state-transition model and decision rules for medication change of PD patients. This approach assured that the model fulfils the following important characteristics: completeness (it provides outputs for any possible inputs), robustness (it works even if some input data is missing), consistency (the model is free of logical errors), transparency (the model is fully “open” for the inspection of contained decision rules), comprehensibility (the embedded decision rules are easy to understand and explain). The model assess all combinations of possible medication changes that arise from the state-transition model thus allowing interpretation of several different and yet correct scenarios for medication change for patients that suffer from PD.

The future work in the framework of the PD_manager project will be focused on model evaluation and implementation. We intend to verify and validate the model on (1) real-life examples of medication-change decisions, such as the Parkinson Progression Marker Initiative dataset [9], (2) on real case patient’s scenarios, (3) and in comparison with neurologists from different EU countries. The model will be integrated in the PD_manager m-health platform for Parkinson’s disease management [2].

4. ACKNOWLEDGMENTS

The work of the authors was supported by the PD_manager project, funded within the EU Framework Programme for Research and Innovation Horizon 2020, under grant number 643706.

5. REFERENCES

[1] Gatsios D, Rigas G, Miljković D, Koroušić-Seljak B, Bohanec M, Arredondo MS, Antonini A, Konitsiotis S, Fotiadis DI. 2016. Mhealth platform for Parkinson's disease management. CBHI, *18th International Conference on Biomedicine and Health Informatics, February 25-26, 2016, Dubai, UAE*.

[2] PD_manager: m-Health platform for Parkinson’s disease management. EU Framework Programme for Research and Innovation Horizon 2020, Grant number 643706, 2015–2017, http://www.parkinson-manager.eu/

[3] Bohanec, M, Antonini, A, Banks, A, Fotiadis, D.I, Gasparoli, E, Gatsios, E, Gentile, G, Hovhannisyan, T, Konitsiotis, S, Koutsikos, K, Marcante, A, Mileva Boshkoska, B, Miljković, D, Rigas, G, Tsiouris, K.M, Valmarska, A. Decision support models. PD_manager project, *Deliverable D5.2, 2017*

[4] Bohanec M, Rajkovič V, Bratko I, Zupan B, Žnidaršič M. DEX methodology: Three decades of qualitative multi-attribute modelling. *Informatica* 37, 2013, 49–54.

[5] Bohanec, Marko. *DEXi: Program for multi-attibute decision making user's manual : version 5.00*, (IJS delovno poročilo, 11897). 2015.

[6] Bohanec M, Zupan B, Rajkovič V. 2000. Applications of qualitative multi-attribute decision models in health care, *International Journal of Medical Informatics*; 58- 59:191-205

[7] Šušteršič, O, Rajkovič, U, Dinevski, D, Jereb, E, Rajkovič, V. 2009. Evaluating patients' health using a hierarchical multi- attribute decision model. *Journal of international medical research; 37(5):1646-1654*.

[8] Mileva-Boshkoska, Biljana, Miljković, Dragana, Valmarska, Anita, Gatsios, Dimitros, Rligas, George, Konitsiotis, Spyros, Tsiouris, Kostas M., Fotiadis, Dimitrios I., Bohanec, Marko. 2017. Decision support system for medication change of Parkinson's disease patients : a state-transition model. V: LINDEN, Isabelle (ur.). *Proceedings of the 2017 International Conference on Decision Support System Technology, ICDSST 2017, with a theme on Data, Information and Knowledge Visualisation in Decision Making, 29-31 may 2017, Namur, Belgium*.

[9] Parkinson Progression Marker Initiative: The Parkinson Progression Marker Initiative (PPMI). *Progress in Neurobiology 95(4), 2011, 629–635*.

Designing a Personal Decision Support System for Congestive Heart Failure Management

Marko Bohanec, Erik Dovgan,
Pavel Maslov, Aljoša Vodopija,
Mitja Luštrek
Jožef Stefan Institute
Department of Intelligent Systems,
Department of Knowledge
Technologies
Jamova cesta 39, 1000 Ljubljana,
Slovenia
{marko.bohanec, erik.dovgan,
pavel.maslov, aljosa.vodopija,
mitja.lustrek}@ijs.si

Paolo Emilio Puddu,
Michele Schiariti,
Maria Costanza Ciancarelli
Sapienza University of Rome
Department of Cardiovascular,
Respiratory, Nephrologic and Geriatric
Sciences
Piazzale Aldo Moro 5, Roma 00185,
Italy
{paoloemilio.puddu.
michele.schiariti}@uniroma1.it
mcostanza.ciancarelli@gmail.com

Anneleen Baert,
Sofie Pardaens,
Els Clays
Ghent University
Department of Public Health
De Pintelaan 185 – 4K3, 9000 Gent,
Belgium
{anneleen.baert,
sofie.pardaens,
els.clays}@ugent.be

ABSTRACT

In this paper, we describe the design of the HeartMan Decision Support System (DSS). The DSS is aimed at helping patients suffering from congestive heart failure to better manage their disease. The support includes regular measurements of patients' physical and psychological state using a wristband and mobile device, and providing advice about physical exercise, nutrition, medication therapy, and environment management. In the paper, an overall architecture of the DSS is presented, followed by a more detailed description of the module for physical exercise management.

Categories and Subject Descriptors

H.4.2 [Types of Systems]: Decision support.
J.3 [Life and Medical Sciences]: Medical information systems.

General Terms

Algorithms, Management, Measurement, Design, Human Factors

Keywords

Decision Support System, Personal Health System, Congestive Heart Failure, Physical Exercise, Decision Models

1. INTRODUCTION

Congestive heart failure (CHF) occurs when the heart is unable to pump sufficiently to maintain blood flow to meet the body's needs [1]. Symptoms include shortness of breath, excessive tiredness, and leg swelling. CHF is a common, chronic, costly, and potentially fatal condition [2]. In 2015 it affected about 40 million people globally. In developed countries, around 2% of adults have heart failure, increasing to 6–10% in age over 65.

HeartMan (http://heartman-project.eu/) is a research project funded by the European Union's Horizon 2020 research and innovation programme. The project aims to develop a personal health system to help CHF patients manage their disease. CHF patients have to take various medications, monitor their weight, exercise appropriately, watch what they eat and drink, and make other changes to their lifestyle. The HeartMan system will provide accurate advice on disease management adapted to each patient in a friendly and supportive fashion. The DSS follows the best medical practices [3] and is designed so that it never suggests anything that would harm the patient.

In this paper, we present the design of the HeartMan Decision Support System (DSS), which was finalised in June 2017 [3]. In section 2, we describe the overall functionality and architecture of the system, and define the roles of its modules that address (1) physiological measurements, (2) physical exercise, (3) nutrition, (4) medication, (5) environment management, and (6) management of calendars and plans. In section 3, we focus on the physical exercise module and present its most important components for (1) patients' physical capacity assessment, (2) weekly exercise planning, and (3) daily exercise management.

2. DESIGN OF THE HEARTMAN DSS

The HeartMan DSS aims at providing medical advice to CHF patients using predictive models, clinical care guidelines and expert knowledge. The purpose of a typical DSS is to passively present information to decision makers so that they can make maximally informed decisions. This DSS, however, is intended for patients who have limited medical knowledge and are consequently expected to follow guidelines with little discretion. Because of that, the DSS actively provides advice to patients, although it does offer choice where appropriate. In this way, it belongs to the category of cooperative DSS [4].

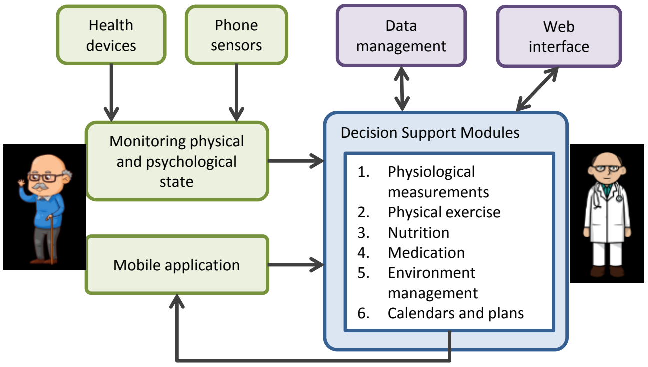


Figure 1: Overall architecture of the HeartMan DSS.

The overall architecture of the HeartMan DSS is shown in Figure 1. The system will use wrist-band sensors to monitor patient's physical activity, heart rate and some other physiological signs. In addition, it will receive data from additional devices, such as scales, smartphone and from the patient via the user interface of the mobile application. This will allow the system to identify the patient's current physical and psychological characteristics. This data will be combined with patient's health data to help them decide on disease control measures. The advice will be tailored to the patient's medical condition by adapting it to the patient's psychological profile (such as normal, poorly motivated, depressed, and anxious) and current health state. The advice will be shown at the mobile app. A web-based interface will be provided to the physician, too, who will be able to monitor the patient's health state and progress, and define or approve parameters that affect the advice given to the patient.

The core of the HeartMan DSS are modules that interpret patient's data and make recommendations. There are six main modules, which address the following aspects of health management:

1. *Physiological measurements*: CHF patients should perform various physiological measurements on a regular basis, such as measuring their weight, blood pressure, heart rate, etc. This raises the patients' awareness of their health, provides valuable information to their physicians, and provides inputs to the DSS. For this purpose, the DSS reminds the patient to regularly perform these measurements, provides the functionality to carry them out, and manage the collected data.

2. *Exercise*: Physical conditioning by exercise training reduces mortality and hospitalization, and improves exercise tolerance and health-related quality of life. For this purpose, the DSS provides a comprehensive exercise programme, which is detailed later in section 3.

3. *Nutrition*: CHF patients should maintain their body weight and take care of their diet, for instance, not eating too much salt or drinking too much fluid. The DSS assesses the patients' nutrition behavior, educates them through a quiz and provides advice towards a healthy diet.

4. *Medication*: Good adherence to medication therapy decreases mortality and morbidity, and improves well-being of CHF patients. For this purpose, the DSS reminds the patient to take medications and assesses the patient's adherence to the medication scheme. For each medication, the patient may obtain an explanation why the adherence is important.

5. *Environment management*: Environmental conditions, such as temperature and humidity, may affect the patient's feeling of health. Combining both, the patient's and environmental conditions, the DSS advises the patient how to change the environment to improve their health feeling.

6. *Calendars and plans*: Given all the DSS aspects (measurement, exercise, nutrition, medication and environment management) and many interactions between them, it is important to sensibly arrange all the activities and notifications, for instance not sending nutrition advice during exercise or suggesting physical exercise after taking diuretics. This DSS module thus coordinates the activities and arranges all the plans into one single calendar.

To date, all these modules have been designed in terms of their functionality, requirements, input data, processing, outputs, and distribution between the client (patient's mobile app) and the server (the DSS in "the cloud") [3].

3. PHYSICAL EXERCISE MODULE

The HeartMan DSS administers a comprehensive exercise programme. At the beginning, the DSS collects medical information and assesses patient's physical capacity in order to plan the difficulty level of the exercises. Then, the DSS provides a weekly set of endurance and resistance exercises, which increase in difficulty as the patient becomes fitter. The DSS also guides the patient during each exercise session: it checks whether the patient is ready to start, then provides instructions, and finally asks the patient to evaluate the exercise. The exercise module follows the guidelines provided in [5] with minor modifications to fit in a mobile application.

3.1 Physical Capacity Assessment

Prior to starting using the HeartMan DSS, the patients should perform a cardiopulmonary exercise (cycloergometry) test to assess their physical capacity. Alternatively, when using the system in a supervised, standardized setting, patients can perform a 6-minute walking test. On this basis, the *physical capacity* of each patient is assessed as "low" (less than 1 W/kg measured by cycloergometry or less than 300 m walked in 6 minutes) or "normal" (otherwise).

3.2 Weekly Exercise Planning

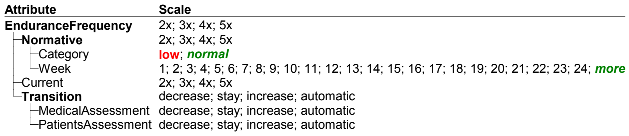
The DSS system provides the patient with a combined *endurance* and *resistance* exercise programme. Both types follow the same principle described with four parameters: *frequency* (times per week), *intensity*, *duration* and *type*. These parameters are combined with the physical capacity to make a *weekly exercise* plan for each patient. For instance, low-capacity patients start with very light 10-15-minute endurance exercises twice per week.

According to the patient's progress, these parameters may change with time. In the HeartMan DSS, the progress is prescribed by two models:

- *EnduranceFrequency*: a model for suggesting weekly frequency of endurance exercises;

- *EnduranceTime*: a model for suggesting weekly time boundaries of endurance exercises.

Both models are formulated using a qualitative multi-criteria decision analysis method DEX [6]. Here, we illustrate the approach describing the *EnduranceFrequency* model, whose structure is shown in Figure 2.



Attribute	Scale
EnduranceFrequency	2x; 3x; 4x; 5x
├**Normative**	2x; 3x; 4x; 5x
│ ├Category	low; ***normal***
│ └Week	1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15; 16; 17; 18; 19; 20; 21; 22; 23; 24; ***more***
├Current	2x; 3x; 4x; 5x
└**Transition**	decrease; stay; increase; automatic
├MedicalAssessment	decrease; stay; increase; automatic
└PatientsAssessment	decrease; stay; increase; automatic

Figure 2: Structure of the *EnduranceFrequency* model.

The *EnduranceFrequency* model is aimed at suggesting the *frequency of exercises* for the next week, based on the patient's physical capacity, week in the programme, current frequency, and the possible physician's and patient's suggestions for the change. In other words, the model takes into account both the normative (as proposed by a general programme) and actual (as practiced by the patient) frequency, leveraging the patient's and physician's opinion about the suggestion for the subsequent week.

The overall recommendation, which is 2, 3, 4, or 5 times per week, is represented by the root attribute *EnduranceFrequency* (Figure 2). The recommendation depends on three sub-criteria:

1. *Normative*: Frequency as suggested according to the default programme. It depends on the patient's physical capacity ("low" or "normal" *Category*) and the current *Week*. The progression is defined by rules presented in Table 1.
2. *Current*: The frequency of exercises currently carried out by the patient; it can run ahead or behind the *Normative* plan. In order to make only small and gradual changes to the frequency, *Current* is compared to *Normative* and only a one-step change is suggested in each week.
3. *Transition* is an attribute that captures the patient's wish and the physician's opinion about changing the frequency. The possible values are "decrease", "same", "increase" or "automatic"; the latter is meant to suggest the frequency according to the normal plan, for instance, when neither the patient or physician have given any suggestion. The patient's and physician's suggestions are combined according to decision rules shown in Table 2. The first two rules say that whenever the patient or the physician suggest to decrease the frequency, it should indeed be decreased (the symbol '*' represents any possible value). Rules 3 and 4 suggest to keep the current frequency whenever one of the participants suggests so, unless the other participant suggests "decrease" Rules 5 and 6 define a similar reasoning for "increase". If both participants have no particular suggestions, the "automatic" transition according to the normal plan takes place.

Table 1: Decision table defining the *Normative* frequency.

	Category	Week	Normative
1	low	<=4	2x
2	low	5–12	3x
3	*normal*	<=6	3x
4	low	13–18	4x
5	*normal*	7–12	4x
6	low	>=19	5x
7	*normal*	>=13	5x

Table 2: Decision rules for *Transition*.

	MedicalAssessment	PatientsAssessment	Transition
1	decrease	*	decrease
2	*	decrease	decrease
3	stay	not decrease	stay
4	not decrease	stay	stay
5	increase	increase or automatic	increase
6	increase or automatic	increase	increase
7	automatic	automatic	automatic

3.3 Daily Exercise Management

Once a weekly plan has been established, the HeartMan DSS assists the patient in carrying out their daily exercises. This consists of four activities: (1) reminding the patient, (2) pre-exercise checking, (3) exercise monitoring, and (4) post-exercise assessment.

3.3.1 *Reminding the patient*

Patients can choose the days when they want to exercise (e.g., every Tuesday, Thursday and Sunday). On these particular days, the patients are at some predefined morning time reminded about the daily exercise. Another reminder is issued if the exercise has not been completed before the given afternoon time.

3.3.2 *Before the exercise*

Before the start of each exercise session, the HeartMan DSS checks if all prior-exercise requirements are met, and advises the patients about safety. Figure 3 shows the decision model.

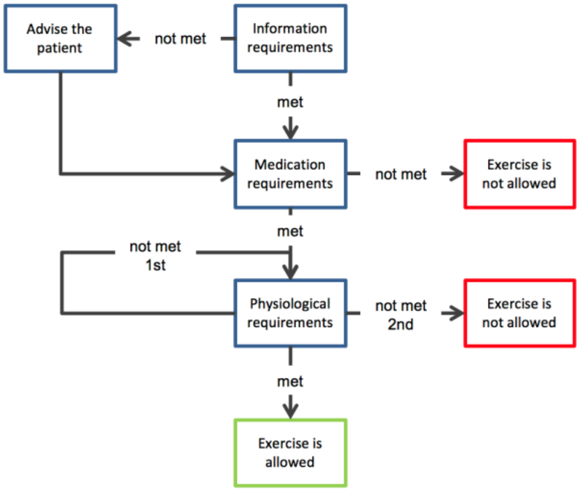
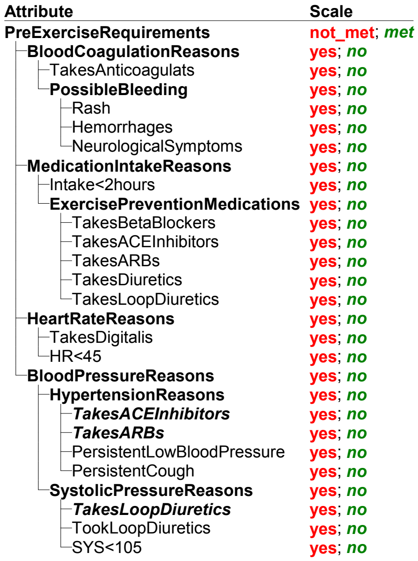


Figure 3: Pre-exercise assessment.

1. *Information requirements:* The blood pressure should have been measured during the day. If not, the patients are instructed to measure it. The pre-exercise heart rate is measured automatically by the wristband; the system makes sure that it is actually worn.
2. *Medication requirements*: Patients are asked to fill a check-list of frequently seen side effects based on their medication schemes and symptoms (e.g., dizziness and chest pain). On this basis the DSS checks for any possible restrictions due to medications or symptoms and suggests rescheduling the session if necessary. The physician or nurse are contacted if severe side effects are present. In the case of dizziness or chest pain, patients are instructed to rest until the symptoms are no longer present.
3. *Physiological requirements*: If all the requirements are met, patients can start with the exercise, otherwise they are instructed to repeat the measurements after five minutes of rest. If after re-checking the measurements are still not within safe limits, exercise is not allowed and patients are advised to contact their physician or heart failure nurse.

Again, a DEX [3, 6] model is employed for assembling and checking the medical conditions, which include medical intake, comorbidities and current physical condition of the patient. The structure and scales of the *PreExerciseRequirements* model are shown in Figure 4. All attributes are binary ("yes"/"no" or "not_met"/"met"). The values of the input attributes are determined from patient data whenever the pre-exercise requirements are checked (normally once per day before making exercises). The subtrees of the model comprise four main groups of possible reasons against participating in the exercises:

- *Blood coagulation*: Whenever the patient takes anticoagulants and there are symptoms indicating a possible bleeding: rash, hemorrhages, or neurological symptoms.
- *Medication intake*: Whenever one of the following medications has been taken 2 hours or less before the exercise: beta blockers, ACE inhibitors, ARBs, diuretics, or loop diuretics.
- *Heart rate*: Whenever the patient takes Digitalis and his/her HR is less than 45 bpm.
- *Blood pressure*: Whenever there are risks of hypertension (taking ACE inhibitors or ARBs, and the patient has persistent low blood pressure or persistent cough) or problems regarding the systolic blood pressure (when the patient's systolic blood pressure is less than 105 and he/she recently took loop diuretics).



Attribute	Scale
PreExerciseRequirements	not_met; *met*
BloodCoagulationReasons	yes; *no*
TakesAnticoagulats	yes; *no*
PossibleBleeding	yes; *no*
Rash	yes; *no*
Hemorrhages	yes; *no*
NeurologicalSymptoms	yes; *no*
MedicationIntakeReasons	yes; *no*
Intake<2hours	yes; *no*
ExercisePreventionMedications	yes; *no*
TakesBetaBlockers	yes; *no*
TakesACEInhibitors	yes; *no*
TakesARBs	yes; *no*
TakesDiuretics	yes; *no*
TakesLoopDiuretics	yes; *no*
HeartRateReasons	yes; *no*
TakesDigitalis	yes; *no*
HR<45	yes; *no*
BloodPressureReasons	yes; *no*
HypertensionReasons	yes; *no*
TakesACEInhibitors	yes; *no*
TakesARBs	yes; *no*
PersistentLowBloodPressure	yes; *no*
PersistentCough	yes; *no*
SystolicPressureReasons	yes; *no*
TakesLoopDiuretics	yes; *no*
TookLoopDiuretics	yes; *no*
SYS<105	yes; *no*

Figure 4: Structure of the *PreExerciseRequirements* model.

3.3.3 During the exercise

If the exercise is allowed, a list of exercises is shown to the patient, who can then select the preferred exercise. After selecting the exercise, a detailed description (text or graphical) regarding the exercise is provided.

During the exercise, the heart rate and systolic blood pressure are continuously measured by the wristband. The patients are advised to stop the exercise in case of symptoms or measurements lying outside of prescribed safety margins. If the heart rate is within the safety limits, but too low or too high, the patent is advised to increase or decrease the intensity, respectively. The system also advises the patients about the exercise duration and is capable of recognizing a premature ending.

3.3.4 After the exercise

After completing the exercise, the patients can rate their feeling of intensity (very light, light, moderate, intense, very intense). Then the system assesses the exercise based on measurements recorded during the exercise. It checks if the exercise was prematurely finished and if the intensity was on average in the prescribed limits. The system takes into account this information when assessing the adherence to the exercise plan and the patient's improvement. Independent of this, the exercise is shown as completed and the weekly plan is updated.

4. CONCLUSION

This paper described the design of the HeartMan DSS that is concerned with "medical" interventions (i.e., interventions that try to improve the patients' physical condition as opposed to psychological). The DSS is based on clinical guidelines for the self-management of CHF, additional medical literature and expert knowledge from the project consortium. The DSS is designed in terms of process models (the order of actions and questions) and decision models (how to make some complex decisions – branching in the process models) for five main topics of CHF management: physiological measurements, exercise, nutrition, medication, and environment management.

The DSS is currently being integrated in the overall HeartMan platform. A comprehensive validation involving 120 CHF patients (of whom 40 are controls without the DSS) is planned for 2018.

5. ACKNOWLEDGMENTS

The HeartMan project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 689660. Project partners are Jožef Stefan Institute, Sapienza University, Ghent University, National Research Council, ATOS Spain SA, SenLab, KU Leuven, MEGA Electronics Ltd and European Heart Network.

6. REFERENCES

[1] Heart failure. *Health Information*. Mayo Clinic. 23 December 2009. DS00061. http://www.mayoclinic.org/diseases-conditions/heart-failure/basics/definition/con-20029801.

[2] McMurray, J.J., Pfeffer. M.A. 2005. Heart failure. *Lancet* 365 (9474): 1877–89. DOI= https://doi.org/10.1016%2FS0140-6736%2805%2966621-4.

[3] Bohanec, M., Vodopija, A., Dovgan, W., Maslov, P., Luštrek, M., Puddu, P.E., Clays, E., Pardaens, S., Baert, A. (2017). *Medical DSS for CHF*. HeartMan Deliverable D4.1.

[4] Turban, E., Sharda, R., Delen, D., King, D. (2010). *Business Intelligence*. 2nd Edition, Prentice Hall.

[5] Piepoli, M.F., Conraads, V., Corrà, U., et al. (2011): Exercise training in heart failure: from theory to practice. A consensus document of the Heart Failure Association and the European Association for Cardiovascular Prevention and Rehabilitation. *European Journal of Heart Failure* 13, 347–357.

[6] Bohanec, M., Rajkovič, V., Bratko, I., Zupan, B., Žnidaršič, M. (2013): DEX methodology: Three decades of qualitative multi-attribute modelling. *Informatica* 37, 49–5

Continuous Blood Pressure Estimation from PPG Signal

Gašper Slapničar
Jožef Stefan Institute
Jamova cesta 39
1000 Ljubljana
gasper.slapnicar@ijs.si

Matej Marinko
Faculty of Math. and Physics
Jadranska cesta 19
1000 Ljubljana
matejmarinko123@gmail.com

dr. Mitja Luštrek
Jožef Stefan Institute
Jamova cesta 39
1000 Ljubljana
mitja.lustrek@ijs.si

ABSTRACT

Given the importance of blood pressure (BP) as a direct indicator of hypertension, regular monitoring is encouraged in people and mandatory for such patients. We propose an approach where photoplethysmogram (PPG) is recorded using a wristband in a non-obtrusive way and subsequently BP is estimated continuously, using regression methods based solely on PPG signal features. The approach is validated using two distinct datasets, one from a hospital and the other collected during every-day activities. The best achieved mean absolute errors (MAE) in a Leave-one-subject-out experiment with personalization are as low as 11.87 ± 12.31 / 11.09 ± 9.99 for systolic BP and 5.64 ± 5.73 / 6.18 ± 4.85 for diastolic BP.

Keywords

Photoplethysmography, blood pressure, regression analysis, m-health

1. INTRODUCTION

According to the World Health Organization (WHO), cardiovascular diseases were the most common cause of death in 2015, responsible for almost 15 million deaths combined [1]. Hypertension is a common precursor of such diseases and can be easily detected with regular blood pressure (BP) measurements.

Given the importance of BP, people should actively monitor its changes. This is not trivial as the traditional "gold standard"BP measurement method involves an inflatable cuff, which should be correctly placed directly above the main artery in the upper arm area, at approximately heart height [9]. These requirements impose relatively strict movement restrictions on the patient and require substantial time commitment. Furthermore, when done by the patient himself, the process can cause stress, which in turn influences the BP values, so it is most commonly done by medical personnel. However, when BP is measured by medical personnel, this can again cause anxiety in the patient, commonly known as white coat syndrome.

Our work focuses on analyzing the photoplethysmogram (PPG) and then developing a robust non-obtrusive method for continuous BP estimation, which will be implemented and used in an m-health system, based on a wristband with a PPG sensor.

2. RELATED WORK

Photoplethysmography is a relatively simple and affordable technique, which is becoming increasingly popular in wearables for heart rate estimation. Exploring its applications, we can see that it is also becoming more widely used in BP estimation in one of two common approaches: *1.)* BP estimation from two sensors (PPG + Electrocardiogram (ECG)) or *2.)* BP estimation using PPG only.

PPG is based on illumination of the skin and measurement of changes in its light absorption. It requires a light source (typically a light-emitting diode – LED light) to illuminate the tissue (skin), and a photodetector (photodiode) to measure the amount of light either transmitted or reflected to the photodetector. Thus, PPG can be measured in either transmission or reflectance mode. With each cardiac cycle the heart pumps blood towards the periphery of the body, producing a periodic change in the amount of light that is absorbed or reflected from the skin, as the skin changes its tone based on the amount of blood in it [6].

The first approach suggests the use of two sensors, typically an ECG and a PPG sensor, in order to measure the time it takes for a single heart pulse to travel from the heart to a peripheral point in the body. This time is commonly known as pulse transit time (PTT) or pulse arrival time (PAT) and its correlation with BP changes is well established [2].

The more recent approach is focused on PPG signal only, however the relationship between PPG and BP is only postulated and not well established, unlike the relationship between PTT and BP. This approach is the least obtrusive by far and PPG sensors have recently become very common in most modern wristbands.

One of the earliest attempts at this approach was conducted by Teng et. al. [3] in 2003. The relationship between arterial blood pressure and certain features of the photoplethysmographic (PPG) signals was analyzed. Data was obtained from 15 young healthy subjects in a highly controlled laboratory environment, ensuring constant temperature, no movement and silence. The mean differences between the linear regression estimations and the measured BP were 0.21 mmHg for SBP and 0.02 mmHg for DBP. The corresponding standard deviations were 7.32 mmHg for SBP and 4.39 mmHg.

A paper was published in 2013 in which authors used data from Multiparameter Intelligent Monitoring in Intensive Care (MIMIC) waveform database [4] to extract 21 time domain features and use them as an input vector for artificial neural networks (ANNs). The results are not quite as good as the linear regression model described earlier, however the data is obtained from a higher number and variety of patients in a less controlled environment, but was still measured in a hospital setting and an undisclosed subsample of all available data was taken. The results reached mean absolute difference between the estimation and the ground truth of less than 5 mmHg with standard deviation of less than 8 mmHg [5].

It is clear that the PPG only approach has potential, however a robust method that works well on a general case is yet to be developed.

3. METHODOLOGY

The workflow consists of two main parts, namely the signal pre-processing and machine learning part. In signal pre-processing, our PPG signal is cleaned of most noise and segmented into cycles, where one cycle corresponds to a single heart beat. Afterwards, features are extracted on per-cycle basis and fed into regression algorithms which build models that are further evaluated.

3.1 Signal pre-processing

When PPG is used in a wristband, the main problem comes from the contact between the sensor and the skin. During everyday activity, the patient moves his arm a lot, which in turn causes substantial movement artefacts in the signal. This is partially alleviated by the usage of green light, which is less prone to artefacts, however pre-processing is still required.

3.1.1 Cleaning based on established medical criteria

In first phase, both BP and PPG signal are roughly cleaned based on established medical criteria [8]. During this phase, parts of signals with systolic BP (SBP) > 280mmHg or diastolic BP (DBP) < 20mmHg or the difference between SBP and DBP < 20mmHg, are removed. This removes parts of signals for which the reference BP signal most likely contained an anomaly as such values indicate extreme medical condition and are not feasible in a common patient.

3.1.2 Peak and cycle detection

In order to do further cleaning and feature extraction, PPG cycle detection is mandatory. This is again not trivial, as substantial noise in the PPG signal poses a significant problem.

A slope sum function, which enhances the abrupt upslopes of pulses in the PPG signal is first created. Afterwards, a time-varying threshold for peak detection is applied [7]. After the peaks are detected, finding the cycle start-end indices is rather simple as the valleys between peaks must be found. An example of detected peaks and cycle locations is shown in Figure 1.

Once cycles are detected, they are used for further cleaning and feature extraction.

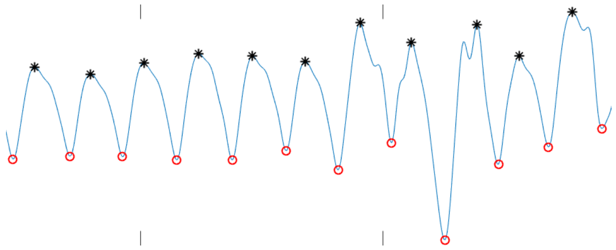


Figure 1: An example output of peak/cycle detection algorithm on PPG signal. Black asterisks correspond to a detected peak while red circles correspond to a detected cycle beginning.

3.1.3 Cleaning based on ideal templates

In the second cleaning phase, a sliding window of 30 seconds is taken and the mean of all cycles within this window is computed from the PPG signal. Presuming that the majority of cycles within a 30sec window are not morphologically altered, a good "ideal cycle template" is created. Each individual cycle is then compared to this ideal template and its quality is evaluated with three signal quality indices (SQIs). The most likely length of cycle L is always determined with autocorrelation analysis. The template is computed by always taking L samples of each cycle in the current window.

Signal quality indices are computed as follows. *SQI1*: First L samples of each cycle are taken, and each cycle is directly compared to the template using a correlation coefficient. *SQI2*: Each cycle is interpolated to length L and then the correlation coefficient is computed. *SQI3*: The distance between template and cycle is computed using dynamic time warping (DTW).

Finally thresholds for each SQI are determined and if more than half cycles in the given 30sec window are discarded, the whole window is considered too noisy and thus removed. Example of this cleaning is shown in Figure 2.

Once high quality signal is obtained, features can be extracted from each cycle.

3.2 Machine learning

In accordance with related work [5] several time domain features were computed and the set of features was further expanded with some from the frequency domain [8]. These are shown in Figure 3.

These features were extracted for each cycle and used in machine learning to derive a regression model for BP estimation.

4. EXPERIMENTS AND RESULTS

In an effort to make our method as general as possible, two datasets were considered for our experiment and all the data which had both PPG and BP signal were used.

4.1 Data

First is the publicly accessible MIMIC database set from which all the patients having both PPG and arterial BP

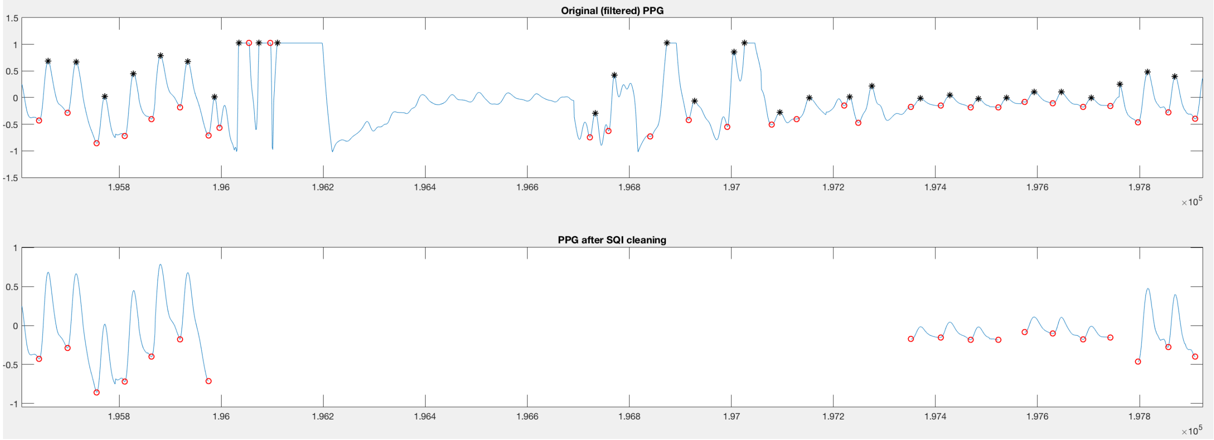


Figure 2: An example of the cleaning algorithm in the 2nd phase. Comparing the top (uncleaned) and bottom (cleaned) signal, we see that the obvious artefact period is discarded.

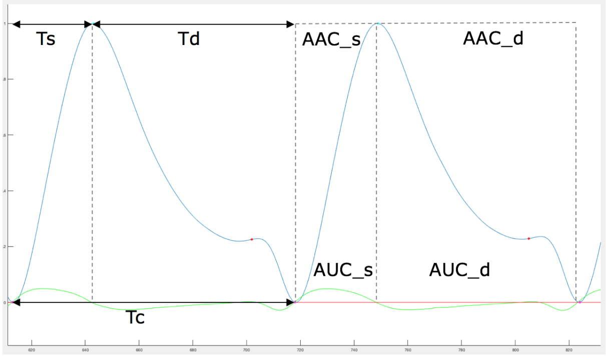


Figure 3: Time domain features which were used. Tc = cycle time, Ts = systolic rise time, Td = diastolic fall time, AAC = area above the curve and AUC = area under the curve for systolic and diastolic part of a cycle.

(ABP) signal were taken. This results in 50 anonymous patients, each having on average several hours of both signals available. The data was collected in a hospital environment, using the hospital equipment.

Second is a dataset collected at Jozef Stefan Institute (JSI) using the Empatica E4 wristband for PPG and an Omron cuff-based BP monitor for the ground truth BP, as is common in related work [3]. The collection procedure was conducted in accordance with recommended clinical protocol [9], ensuring correct placement of the cuff on upper arm with the sensor above the main artery and its location at approximately heart height. The subjects were in a sitting upright position during the measurements, thus following the protocol as best as possible. Ideally, arterial BP would be measured in the artery as ground truth, however due to invasive nature of the procedure, this is not feasible in an everyday life situation, so an upper arm cuff-based digital monitor was used as a good replacement. These devices are superior to wrist cuff-based monitors, as wrist devices are less accurate and extremely sensitive to body position.

In the first phase of data collection, 8 healthy subjects were considered, 5 male and 3 female. Each wore the wristband for several hours during every-day activities and measured their BP every 30min or more often. Finally, only parts of signals 3 minutes before and after the BP measurement were taken into consideration.

4.2 Experimental setup

Leave-one-subject-out experiment was conducted on each dataset, as it is the most suitable experiment to evaluate the generalization performance of the algorithms. Due to time and computational power restrictions, data was subsampled by taking 500 uniformly selected cycles.

During the initial attempt, a regression model was trained in each iteration on all subjects, except the left out. This yielded poor results, hinting at the fact, that most patients are unique in some way. This was confirmed by doing a cycle morphology analysis during which it was established that different subjects have different cycle shapes and that similar cycle shapes do not signify similar BP values. Thus, personalization of the trained models was considered.

In the second attempt, the regression models were again trained using all subjects except the left out, however they were further personalized using some data instances from the left out subject. The instances of the left out subject were first grouped by BP values and these groups were then sorted from lowest to highest BP. Afterwards, every *n-th* group ($n = 2,3,4,5,6$) of instances was taken from the testing data and used in training in order to personalize the model to the current patient. This ensures personalization with different BP values, as taking just a single group of instances gives little information, since the BP will be constant within this group. Given the fact that MIMIC data consists of roughly 5x the amount of patients compared to JSI collected data, the personalization data for it was multiplied 5 times, making it noticable within the large amount of training data from the remaining patients.

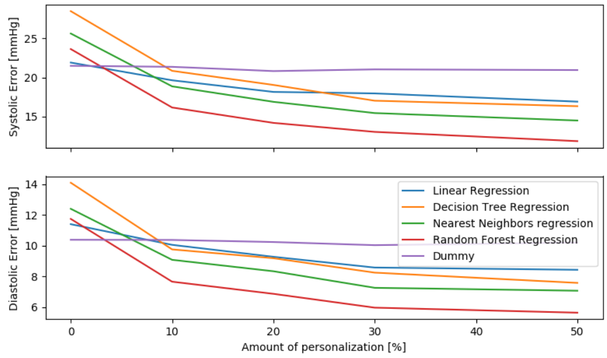


Figure 4: MAE for SBP and DBP for MIMIC dataset at different amounts of personalization.

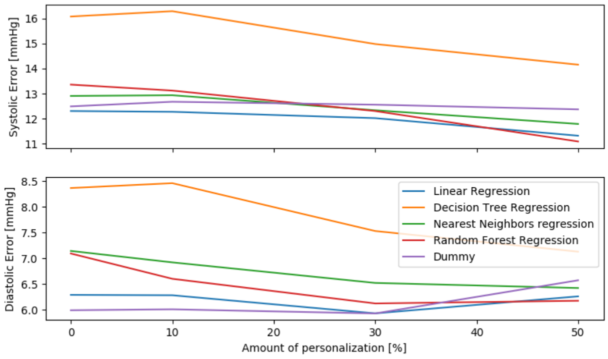


Figure 5: MAE for SBP and DBP for JSI collected dataset at different amounts of personalization.

During both attempts, several regression algorithms were considered, as given in Figures 4 and 5. Mean Absolute Error (MAE) was used as a metric. All models were compared with a dummy regressor, which always predicted the mean BP value of the same combination of general and personalization data as the other models used to train themselves. Finally, the regressor with the lowest MAE was chosen.

For successful personalization, the user should measure his PPG continuously and also make a few periodic measurements of his BP using a reliable commercial device. This allows the model to personalize to the user, learning from a small sample of his labeled data, thus improving its predictive performance.

4.3 Results

Due to low variations in BP, the dummy regressor often performs relatively well, however for MIMIC data with more BP variation, some improvements have been made as shown in Figure 4. The JSI collected data has proven to be more problematic, as there are only a low amount of different BP values in this phase of collection.

The lowest error using MIMIC data was achieved by using the RandomForest regression algorithm, with the highest amount of personalization. The achieved errors were 11.87 ± 12.31 for SBP and 5.64 ± 5.73 for DBP.

Due to high amount of movement artefacts in JSI collected data, a lot of data was removed by the cleaning algorithm, leaving a very low amount of usable data with very low variations in BP. This further enhanced the performance of dummy regressor, while leaving little information for other algorithms. Best achieved errors of 11.09 ± 9.99 for SBP and 6.18 ± 4.85 for DBP are only slightly surpassing the mean predictions at maximum personalization, as shown in Figure 5.

5. CONCLUSION

We have developed a pipeline for BP estimation using PPG signal only and have evaluated its performance on two distinct datasets.

First part of the pipeline does signal pre-processing, removing most movement artefacts and detecting PPG cycles. The second part computes features on per-cycle basis and feeds them in regression algorithms. These were evaluated on hospital collected MIMIC database data as well as field collected data at JSI using a wristband. Due to low variations in subject's BP and high variation in their PPG, there is limited information about the correlation between the two, however promising results were obtained with best achieved mean absolute errors (MAE) in a Leave-one-subject-out experiment with personalization as low as 11.87 ± 12.31 / 11.09 ± 9.99 for systolic BP and 5.64 ± 5.73 / 6.18 ± 4.85 for diastolic BP.

Acknowledgement

The HeartMan project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 689660. Project partners are Institut Jozef Stefan, Unversita Degli Studi Di Roma La Sapienza, Universiteit Gent, Consiglio Nazionale Delle Richerche, ATOS Spain SA, SenLab, KU Leuven, MEGA Elektronikka Oy and European Heart Network.

6. REFERENCES

[1] The World Health Organization. "The top 10 causes of death", 2015.

[2] Geddes et. al. "Pulse transit time as an indicator of arterial blood pressure", 1981.

[3] Teng et. al. "Continuous and noninvasive estimation of arterial blood pressure using a photoplethysmographic approach", 2003.

[4] Goldberger et. al. "PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals", 2000.

[5] Lamonaca et. al. "Application of the Artificial Neural Network for blood pressure evaluation with smartphones", 2013.

[6] Allen. "Photoplethysmography and its application in clinical physiological measurement", 2007.

[7] Lázaro et. al. "Pulse Rate Variability Analysis for Discrimination of Sleep-Apnea-Related Decreases in the Amplitude Fluctuations of Pulse Photoplethysmographic Signal in Children", 2014.

[8] Xing et. al. "Optical Blood Pressure Estimation with Photoplethysmography and FFT-Based Neural Networks", 2016.

[9] Frese, E. M. and Fick, Ann and Sadowsky, H. S. "Blood Pressure Measurement Guidelines for Physical Therapists", 2011

Recognizing Hand-Specific Activities with a Smartwatch Placed on Dominant or Non-dominant Wrist

Božidara Cvetković, Vid Drobnič, Mitja Luštrek
Jožef Stefan Institute
Department of Intelligent Systems
Jamova cesta 39
Ljubljana
boza.cvetkovic@ijs.si

ABSTRACT

In this paper we analyze the use of accelerometer-equipped smartwatch to recognize hand-specific activities. We start with a large set of activities, and since many activities have a similar acceleration pattern, we gradually group semantically similar activities to find a tradeoff between the accuracy on one hand, and semantically understandable and useful activity groups on the other hand. Additionally, we compare the activity recognition in terms of the number of activities and accuracy when wearing a smartwatch on the dominant or non-dominant wrist. The preliminary results show that we can recognize up to seven groups of activities with the dominant, and up to five activity groups with the non-dominant wrist.

Categories and Subject Descriptors

D.3.3 [**Human-centered computing**]: Ubiquitous and mobile computing

Keywords

Activity recognition, wrist wearable, machine learning, accelerometers

1. INTRODUCTION

Activity recognition is an important module in person oriented intelligent systems, since most of the further reasoning or assistance to the user depends on the user's current or past activity. This dependency is highly significant in applications intended for the management of lifestyle and sports activities [6], as well as chronic diseases such as diabetes or chronic heart failure (CHF). In diabetes, the user needs to monitor two particular activities, the eating (which increases the blood glucose level) and exercise (which decreases the blood glucose) [1] and in CHF it is important to monitor the food intake (eating) as well as exercise in terms of its intensity and amount of rest [4].

Due to importance of activity recognition and availability of accelerometer equipped wearables it is not surprising that the research area is very popular and partially also very mature. The maturity of the area is shown in the amount of applications and wearable devices dedicated to activity monitoring available on the market [2, 3]. However, these applications and devices mostly recognize three activities (walking, running and rest), which is insufficient for applications in which e.g., eating or any other hand-oriented activities are important.

In this paper we analyze and evaluate a possibility to use accelerometer equipped smartwatch to recognize a large set of hand-oriented activities. Since many activities have similar pattern we gradually group semantically similar activities into single activity group to find a tradeoff between accuracy and semantically understandable activity groups. Additionally, we compare the activity recognition in terms of number of activities and accuracy when wearing a smartwatch on dominant or non-dominant wrist.

The paper is structured as follows. Section 2 presents the related work on activity recognition, Section 3 introduces the dataset and methods for preprocessing and training the models. The evaluation results are present in Section 4 and Section 5 concludes the paper.

2. RELATED WORK

Pioneers in activity recognition research studied use of single or multiple accelerometers attached to different locations on the users body. Attal et al. [5] reviewed the research done until 2015 and proved that number of recognized activities increases with the number of sensors attached to the users body. Since using one or more dedicated accelerometers was perceived as unpractical, the researchers started using devices that most people already have or will have in the future, such as smartphones and wristbands.

Research on activity recognition with the smartphone mostly covers analysis of accelerometer signals without any knowledge of its orientation, thus recognizing only small fraction of activities (walking, running, rest, etc.) [11]. Martin et al. [10] was first to take varying orientation and location into consideration. Their approach requires use of all available smartphone sensors to estimate the location and normalize the orientation. In our recent research [8], we proposed a real-time method that normalizes the orientation, detects the location and afterward uses a location specific machine-learning model for activity recognition.

The research on activity recognition with wrist-worn devices has started with the accelerometer placed on a persons wrist [5]. Since this is the most comfortable placement of the sensor, the research became popular for recognizing sports activities [12] and common activities (sitting, standing, lying, walking, running) [7]. However, none of the research focused on recognizing hand-specific activities (e.g., eating, washing, hammering, etc.), which is the topic of this paper.

3. MATERIALS AND METHODS

3.1 Dataset

Dataset contains data of 11 volunteers equipped with two smartwatches with accelerometer and a heart rate sensor (one on each wrist), performing a predefined scenario. Average accelerometer sampling rate was 48.2 Hz (±4.4) for the left hand and 51.3 Hz (±14.2) for the right hand.

The scenario contained 39 different activities, but not all were performed by each volunteer. Figure 1 presents the distribution of data in terms of number of learning instances in the dataset and in terms of people performing the activity (see Section refpreprocess). We collected approximately two hours of data per person. We can observe that some activities were performed by one person only, which is insufficient for training the models and evaluating them using leave-one-subject-out approach. Omitting these activities left us with 30 activities, due to errors in data collection we also had to omit the mobile use, phone call, clapping, white board and rolling dice. This left us with 25 activities for further analysis.

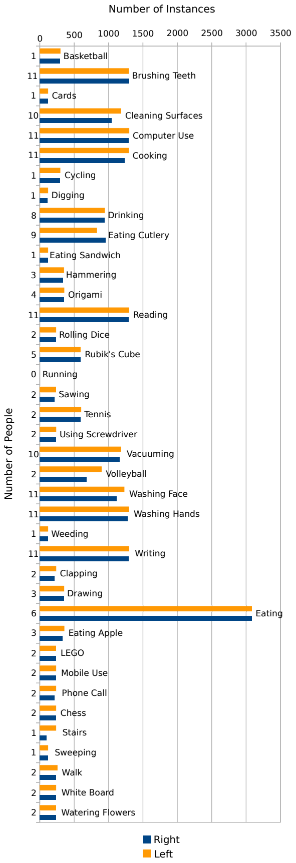


Figure 1: Number of instances per activity (y axis) and number of people performing the activity (x axis)

3.2 Preprocessing

The goal of the preprocessing procedure is to combine the accelerometer and heart rate data received from the smartwatch into form suitable for further use with machine-learning algorithms (feature vectors).

The raw acceleration and heart rate data is first segmented into 2-second windows, each next overlapping by half of its size, from which we extract 90 acceleration features and 4 heart rate features. In brief, the raw acceleration data is first filtered (low-pass and band-pass) to remove noise and gravity. The data is then used for calculation of physical (e.g., velocity, kinetic energy, etc.), statistical features (e.g., the mean, variance, etc.) and features based on signal processing expert knowledge (e.g., number of peaks in a signal, etc.). The reader is referred to [8] for more details about the feature extraction. Once the features are extracted, they are used to form a feature vector to be used for machine-learning.

3.3 Method

Activity recognition is set as a classification task, performed in real-time. The feature vector formed during the feature extraction (Section 3.2) is feed into a classification machine-learning model trained to recognize the activities.

The collected dataset contains data labeled with 25 activities for each wrist. To design an accurate classifier, we had to solve two challenges: (i) the difference in movement of dominant and non-dominant hand during the same activity (e.g., drinking, eating, writing, etc.), and (ii) similar hand movement when performing different activities. We decided to develop two classification models, one for each wrist according to dominance to solve the first challenge. For the second challenge we analyzed the possibility to semantically group the activities, thus achieve higher recognition accuracy but still keep understandability of the recognized activity.

To select the machine-learning classification algorithm to be used for training the models, we have first evaluated the classification accuracy of five different machine-learning algorithms as implemented in Weka suite [9] (J48, SVM, JRip, Random Fores and Naïve Bayes) on the dataset with 25 activities. All experiments are done with Leave-One-Subject-Out approach (LOSO). As in our previous activity monitoring research, the Random Forest achieved the best results and was chosen for all further experiments.

Once the machine-learning algorithm was chosen we analyzed the possible grouping of the activities according to dominance. We started with the dominant hand, the grouping of which is presented in Figure 2. We start by gradually grouping the most similar activities together and evaluating the impact on accuracy. We first group the activities that seem the most similar. All upper hand movements used in face hygiene are grouped together, next are the eating activities, sports and activities similar to writing. The final three groups are the activities where the person plays games or the hand gesture is of low intensity. We also tried to group home chores into low and high intensity, which turned out less accurate then if grouping all home chores together. With this approach we divided all 25 activities into 7 groups or classes to be recognized when smartwatch is worn on dominant hand. Results of each iteration is presented in Section 4.

The same approach was used to group activities to be recognized by non-dominant hand (Figure 3). We grouped the sports activities, eating activities, all chores activities to-

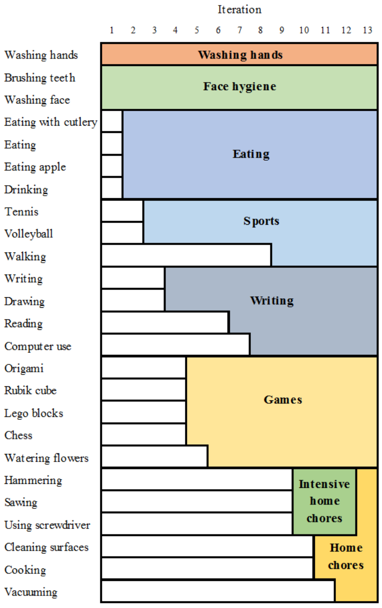


Figure 2: Grouping of activities when smartwatch is worn on a dominant hand.

gether. The activities that were left were very similar in terms of non-dominant hand movement. We tried to distinguish between activities which are similar to writing and games activities, but this decreased the accuracy compared to grouping the two types of activities into single group (hand work). The last group of activities contains the washing activities. With this approach we divided all 25 activities into 5 groups or classes to be recognized when smartwatch is worn on non-dominant hand. The results of each iteration are presented in Section 4

Apart from evaluating the classification models for each wrist on dedicated grouping of activities, we have also evaluated the use of non-dominant hand activities grouping for training the dominant hand model and vice-versa (denoted as Cross). Both experiments were preformed in two ways:

- By using the machine-learning model trained for the specific wrist directly, namely Default approach (D)
- By smoothing the results using the majority classification in the 10-class sliding window, namely smoothing approach (S). The length of the window was selected arbitrarily.

The results are presented in Section 4.

4. EVALUATION

The goal of the evaluation was to analyze and compare the recognition of the activities according to the retrieved data from the smartwatch worn on the dominant or non-dominant wrist. Additionally, we wanted to evaluate and get an insight into type of activities that can be recognized in respect to the hand dominance. The evaluation was performed with Random Forest algorithm in leave-one-subject out (LOSO) manner on dataset presented in Section 3.1. The results are presented in Table 1.

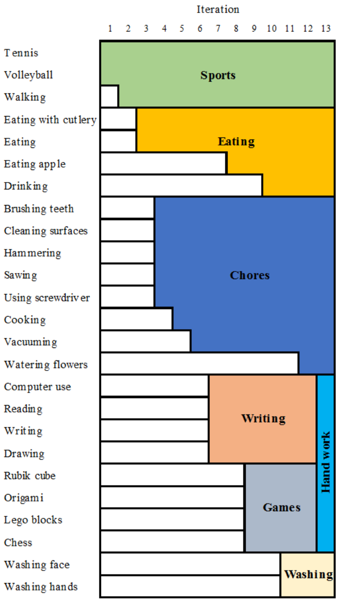


Figure 3: Grouping of activities when smartwatch is worn on a non-dominant hand.

First, we evaluated the use of acceleration and heart rate data retrieved from the smartwatch attached to the dominant wrist. We used the default approach (D) introduced in Section 3.3 to evaluate each grouping of the activities. The increase in accuracy while gradually decreasing the number of recognized classes from 25 to seven is presented in Figure 4. As expected the accuracy increased with each subsequent grouping and we have finally settled for seven classes (Dominant wrist (D)). If we apply smoothing (Dominant wrist (S)) we gain 3 percentage points in accuracy. Finally, we evaluated the recognition of seven classes with non-dominant wrist data, which returned poor accuracy of 58% when default (D) method was used and 63.7% when smoothing was applied (Figure 4 Cross: Non-dominant wrist (S)).

The same approach was used to define the classes to be recognized with non-dominant hand. We first used the default approach (D) on each grouping of the activities which resulted in five final classes. The process of grouping and respective accuracy is presented in Figure 4 (Non-dominant wrist (D)). When smoothing is applied (Non-dominant wrist (S)) we gain 4 percentage points in accuracy. Finally, we evaluated the recognition of five classes with dominant wrist data, which as expected returned higher accuracy then with seven classes (76% when default (D) method was used and 84% when smoothing was applied (Figure 4 Cross: Dominant wrist (S)).

Table 1: Evaluation of activity recognition. The methods: D=default, S=smoothed.

Wrist (method)	Accuracy [%]	# classes
Dominant (D)	71	7
Dominant (S)	79	7
Cross: Non-dominant (D)	58	7
Cross: Non-dominant (S)	64	7
Non-Dominant (D)	70	5
Non-Dominant (S)	74	5
Cross: Dominant (D)	76	5
Cross: Dominant (S)	84	5

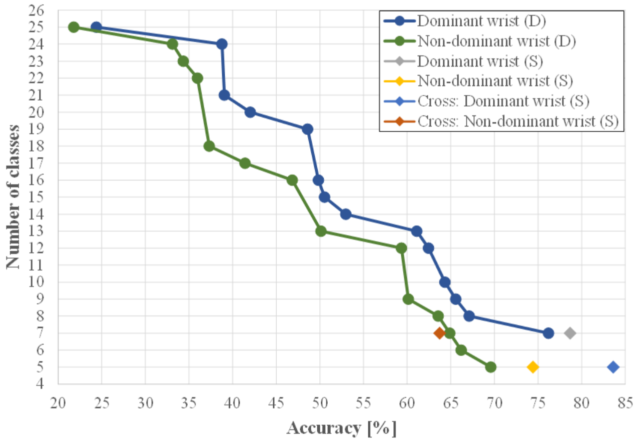


Figure 4: The grouping of activities and accuracy.

5. CONCLUSION

We presented a feasibility study of recognizing hand-specific activities using data retrieved from smartwatch on dominant or non-dominant wrist. We start with large set of hand-specific activities and gradually decrease the number of activities by semantically grouping them together. The preliminary results show that we can recognize larger set of activity groups if we use data from the smartwatch worn on dominant wrist (7 activity groups) then using data from the smartwatch worn on non-dominant wrist (5 activity groups).

Since these are only preliminary results, which gave us a feasibility insight, we will need to repeat the data collection procedure to collect more samples of already recorded activities as well as record additional activities (e.g., sport-specific, home-chores specific, etc.). To achieve higher accuracy, we will also need to perform feature selection procedure and analyze which features are relevant for the task. Finally, we will need to merge the dataset with other datasets that contain non-hand-specific activities and probably design more complex algorithm to achieve good results.

6. ACKNOWLEDGMENTS

The HeartMan project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 689660. Project partners are Jožef Stefan Institute, Sapienza University, Ghent University, National Research Council, ATOS Spain SA, SenLab, KU Leuven, MEGA Electronics Ltd and European Heart Network. We would also like to thank student Grega Mezič for help with the data collection.

7. REFERENCES

[1] American Diabetes Association. `http://www.diabetes.org/food-and-fitness/`. [Online; accessed September-2017].

[2] FitBit. `https://www.fitbit.com/eu/home`. [Online; accessed September-2017].

[3] Runkeeper. `https://runkeeper.com/`. [Online; accessed September-2017].

[4] P. A. Ades, S. J. Keteyian, G. J. Balady, N. Houston-Miller, D. W. Kitzman, D. M. Mancini, and M. W. Rich. Cardiac Rehabilitation Exercise and Self-Care for Chronic Heart Failure, 2013.

[5] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou, and Y. Amirat. Physical Human Activity Recognition Using Wearable Sensors. *Sensors (Basel, Switzerland)*, 15(12):31314–38, 2015.

[6] S. Chatterjee and A. Price. Healthy Living with Persuasive Technologies: Framework, Issues, and Challenges. *Journal of the American Medical Informatics Association*, 16(2):171–178, 2009.

[7] S. Chernbumroong and A. S. Atkins. Activity classification using a single wrist-worn accelerometer. *2011 5th International Conference on Software, Knowledge Information, Industrial Management and Applications (SKIMA) Proceedings*, pages 1–6, 2011.

[8] B. Cvetković, R. Szeklicki, V. Janko, P. Lutomski, and M. Luštrek. Real-time activity monitoring with a wristband and a smartphone. *Information Fusion*, 2017.

[9] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The WEKA data mining software. *SIGKDD Explorations Newsletter*, 11(1):10, 2009.

[10] H. Martín, A. M. Bernardos, J. Iglesias, and J. R. Casar. Activity logging using lightweight classification techniques in mobile devices. *Personal and Ubiquitous Computing*, 17(4):675–695, 2013.

[11] M. Shoaib, S. Bosch, O. Incel, H. Scholten, and P. Havinga. A Survey of Online Activity Recognition Using Mobile Phones. *Sensors*, 15(1):2059–2085, 2015.

[12] P. Siirtola, P. Laurinen, E. Haapalainen, J. Röning, and H. Kinnunen. Clustering-based activity classification with a wrist-worn accelerometer using basic features. In *2009 IEEE Symposium on Computational Intelligence and Data Mining, CIDM 2009 - Proceedings*, pages 95–100, 2009.

R-R vs GSR – An inter-domain study for arousal recognition

Martin Gjoreski
Department of Intelligent Systems,
Jožef Stefan Institute
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
martin.gjoreski@ijs.si

Blagoj Mitrevski
Faculty of Computer Science and
Engineering
Skopje, R. Macedonia

Mitja Luštrek, Matjaž Gams
Department of Intelligent Systems,
Jožef Stefan Institute
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia

ABSTRACT

Arousal recognition is an important task in mobile health and human-computer interaction (HCI). In mobile health, it can contribute to timely detection and improved management of mental health, e.g., depression and bipolar disorders, and in HCI it can enhance user experience. However, which machine-learning (ML) methods and which input is most suitable for arousal recognition, are challenging and open research questions, which we analyze in this paper.

We present an inter-domain study for arousal recognition on six different datasets, recorded with twelve different hardware sensors from which we analyze galvanic skin response (GSR) data from GSR sensors and R-R data extracted from Electrocardiography (ECG) or blood volume pulse (BVP) sensors. The data belongs to 191 different subjects and sums up to 150 hours of labelled data. The six datasets are processed and translated into a common spectro-temporal space, and features are extracted and fed into ML algorithms to build models for arousal recognition. When one model is built for each dataset, it turns out that whether the R-R, GSR, or merged features yield the best results is domain (dataset) dependent. When all datasets are merged into one and used to train and evaluate the models, the R-R models slightly outperformed the GSR models.

Keywords

Arousal recognition; GSR; R-R; machine learning; health.

1. INTRODUCTION

The field of affective computing [1] has been introduced almost two decades ago and yet modeling affective states has remained a challenging task. Its importance is mainly reflected in the domain of human-computer interaction (HCI) and mobile health. In the HCI, it enables a more natural and emotionally intelligent interaction. In the mobile health, it contributes to the timely detection and management of emotional and mental disorders such as depression, bipolar disorders and posttraumatic stress disorder. For example, the cost of work-related depression in Europe, was estimated to €617 billion annually in 2013. The total was made up of costs resulting from absenteeism and presenteeism (€272 billion), loss of productivity (€242 billion), health care costs of €63 billion and social welfare costs in the form of disability benefit payments (€39 billion) [2].

Affective states are complex states that results in psychological and physiological changes that influence behaving and thinking [3]. These psycho-physiological changes can be captured by a wearable device equipped with GSR, ECG or BVP sensor. For example, the emotional state of fear usually initiates rapid heartbeat, rapid breathing, sweating, and muscle tension, which are physiological signs that can be captured using wearables.

The affective states can be modeled using a discrete or a continuous approach. In the discrete approach, the affect (emotions) is represented as discrete and distinct state, i.e., anger, fear, sadness, happiness, boredom, disgust and neutral. In the continuous approach, the emotions are represented in 2D or 3D space of activeness, valance and dominance [3]. Unlike the discrete approach, this model does not suffer from vague definitions and fuzzy boundaries, and has been widely used in affective studies [4] [5] [6]. The use of the same annotating model allows for an inter-study analysis.

In this study we examine arousal recognition from GSR and heart–related physiological data, captured via: chest-worn ECG and GSR sensors, finger-worn BVP sensor, and wrist-worn GSR sensor and pulse oximeter (PPG) sensor. The data belongs to six publicly available datasets for affect recognition, in which there are 191 different subjects (70 females) and nearly 150 hours of arousal-labelled data.

All of this introduces the problem of inter-domain learning, to which ML techniques are sensitive. To overcome this problem, we use preprocessing techniques to translate the datasets into a common spectro-temporal space of R-R and GSR data. After the preprocessing, R-R and GSR features are extracted and are fed into ML algorithms to build models for arousal recognition. Finally, the results between different experimental setups are compared, i.e., models that use only R-R features, models that used only GSR features and models that use both R-R and GSR features. This comparison is performed in a dataset-specific setup and merged setup where all datasets are merged in one. At the end, the experimental results are discussed and the study is concluded with remarks for further work.

RELATED WORK

Affect recognition is an established computer-science field, but many remaining challenges. Many studies confirmed that affect recognition can be performed using speech analysis [21], video analysis [8], or physiological sensors in combination with ML. The majority of the methods that use physiological signals use data from ECG, electroencephalogram (EEG), functional magnetic resonance imaging (fMRI), GSR, electrooculography (EOG) and/or BVP sensors.

In general, the methods based on EEG data outperform the methods based on other data [4] [5], probably due to the fact the EEG provides a more direct channel to one's mind. However, even though EEG achieves the best results, it is not applicable in normal everyday life. In contrast, affect recognition from R-R intervals or GSR data, is much more unobtrusive since this data can be extracted from ECG sensors, BVP sensors, PPG or GSR sensors, most of which can be found in a wrist device (e.g., Empatica [9] and Microsoft Band [10]). Regarding the typical ML approaches for affect recognition, Iacoviello et al. have combined discrete wavelet transformation, principal component analysis and support vector machine (SVM) to build a hybrid classification framework using EEG [11]. Khezri et al. used EEG combined with GSR to recognize six basic emotions via K-nearest neighbors (KNN) classifiers [12]. Verma et al. [13] developed an ensemble approach using EEG, electromyography (EMG), ECG, GSR, and EOG. Mehmood and Lee used independent component analysis to extract emotional indicators from EEG, EMG, GSR, ECG, and (effective refractory period) ERP [14]. Mikuckas et al. [15] presented a HCI system for emotional state recognition that uses spectro-temporal analysis only on R-R signals. More specifically, they focused on recognizing stressful states by means of the heart rate variability (HRV) analysis.

However, a clear comparison between ML methods for affect recognition from unobtrusively captured sensor data (e.g., R-R vs. GSR data) has not been presented yet, since most of these studies focused on only one dataset and a combination of the sensor data, aiming towards the highest performance and disregarding the obtrusiveness of the system. In this work, we analyze which ML algorithms in combination with which data type (either R-R intervals or GSR) would yield best performance across six different datasets (domains) for arousal recognition.

2. DATA

The data belongs to six publicly available datasets for affect recognition: Ascertain, Deap, Driving workload dataset, Cognitive load dataset, Mahnob, and Amigos. Overall, nearly 150 hours of arousal-labelled data that belong to 191 subjects. Table 1 presents the number of subjects per dataset, the mean age, number of trials per subject, mean duration of each trial, duration of data per subject - in seconds, and overall duration.

Table 1. Experimental data summary.

Dataset	Subjects	Females	Mean age	Trials	Duration per		
					trial [s]	subject [min]	dataset [h]
Ascertain	58	21	31	36	80	48.0	46.4
DEAP	32	16	26.9	40	60	40.0	21.3
Driving	10	3	35.6	1	1800	30.0	5.0
Cognitive	21	0	28	2	2400	80.0	28.0
Mahnob	30	17	26	40	80	53.3	26.7
Amigos	40	13	28	16	86	22.9	15.3
Overall	191	70	29.25	135	884.0	251.3	142.7

The four datasets, Ascertain, Deap, Mahnob and Amigos, were already labelled with the subjective arousal level. One difference between these datasets was the arousal scale used for annotating. For example, the Ascertain dataset used 7-point arousal scale, whereas the Deap dataset used 9-point arousal scale (1 is very low, and 9 is very high). From the both scales, we split the labels in the middle, which is the same split used in the original studies. Similar step was performed for the Mahnob dataset. The two datasets, Driving workload and Cognitive load, did not contain labels for subjective arousal level. The Driving workload dataset was labelled with subjective ratings for a workload during driving session. For this dataset, we presume that increased workload corresponds to increased arousal. Thus, we used the workload ratings as an arousal ratings. The split for high arousal was put on 60%. Similarly, the cognitive load dataset was labelled for subjective stress level during stress inducing cognitive load tasks (mathematical equations). The subjective scale was from 0 to 4 (no stress, low, medium and high stress). We put the limit for high arousal on 2 (medium stress).

3. METHODS

3.1 Pre-processing and feature extraction

3.1.1 R-R data

The preprocessing is essential, since it allows merging of the six different datasets. For the heart-related data, it translates the physiological signals (ECG or BVP) to R-R intervals and performs temporal and spectral analysis. First, a peak detection algorithm is applied to detect the R-R peaks. Next, temporal analysis, i.e., calculating the time distance between the detected peaks, detects the R-R intervals. Once the R-R intervals are detected they can be analyzed as a time-series. First, each R-R signal is filtered using median filter. After the median filter, person specific winsorization is performed with the threshold parameter of 3 to remove outlier R-R intervals. From the filtered R-R signals, periodogram is calculated using the Lomb-Scargle algorithm [7]. The Lomb-Scargle algorithm is used for spectral analysis of unequally spaced data (as are the R-R intervals). Finally, the following HRV features were calculated from the time and spectral representation of the R-R signals: meanHR, meanRR, sdnn, sdsd, rmssd, pnn20, pnn50, sd1, sd2, sd1/sd2, lf, hf, lf/hf [29].

3.1.2 GSR data

To merge the GSR data, several problems were addressed. Each dataset is recorded with different GSR hardware, thus the data can be presented in different units and different scales. To address this problem, each GSR signal was converted to μS (micro Siemens). Next, to address the inter-participant variability of the signal, person-specific min-max normalization was performed, i.e., each signal was scaled to [0, 1] using person specific winsorized minimum and maximum values. The winsorization parameter was set to 3. Finally, the GSR signal was filtered using lowpass filter with a cut-off frequency of 1HZ.

The filtered GSR signal was used to calculate the following GSR features: mean, standard deviation, 1st and 3rd quartile (25th and 75th percentile), quartile deviation, derivative of the signal, sum of the signal, number of responses in the signal, rate of responses in the signal, sum of the responses, sum of positive derivative, proportion of positive derivative, derivative of the tonic component of the signal, difference between the tonic component and the overall signal [21].

3.2 Machine learning

After the feature extraction, every data entry consists of 16 R-R features and 14 GSR features, which can be input for typical ML algorithms. Models were built using seven different ML algorithms: Random Forest, Support Vector Machine, Gradient Boosting Classifier, and AdaBoost Classifier, KNN Classifier, Gaussian Naive Bayes and Decision Tree Classifier. The algorithms were used as implemented in the Scikitlearn, the Python ML library [33]. For each algorithm, randomized search on hyper parameters was performed on the training data using 2-fold validation.

4. EXPERIMENTAL RESULTS

Two types of experiments were performed: dataset specific experiments, and experiments with merged datasets. The evaluation was performed using trial-specific 10-fold cross-validation, i.e., the data segments that belong to one trial (e.g., one affective stimuli), can either belong only to the training set or only to the test set, thus there was no overlapping between the training and test data.

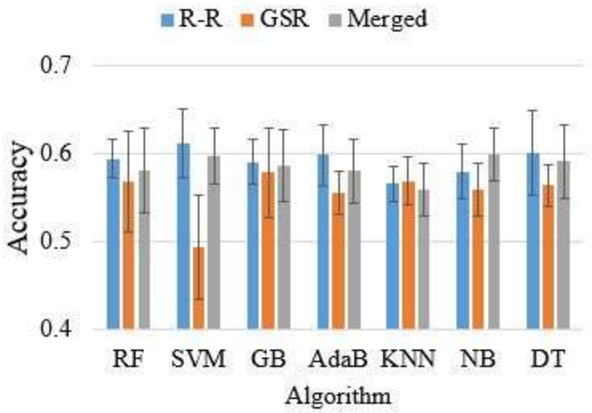
4.1 Dataset specific

The results for the dataset specific experiments are presented in Table 2. The first column represents the ML algorithm, the second column represents the features used as input to the algorithm (R-R, GSR or Merged - M) and the rest of the columns represent the dataset which is used for training and evaluation using the trial-dependent 10-fold cross-validation. We report the mean accuracy ± the standard evaluation for the 10 folds. For each dataset, the best performing model(s) is(are) marked with green. For example, on the Ascertain and the Driving workload dataset, the best performing algorithm is the SVM, on the Deap dataset the best performing algorithm is the RF, on the Cognitive Load and the Mahnob datasets the best performing is the NB, and on the Amigos dataset the best performing is the AdaBoost algorithm.

When we compare which input (R-R features, GSR features or Merged-M) provide better accuracy, on two datasets (the Asceratin and the Driving workload) the results are the same, on the Deap dataset, the R-R features provide better results, on the Cognitive Load dataset the highest accuracy is achieved both for the GSR and the Merged features, on the Mahnob dataset the GSR features provide best accuracy and on the Amigos dataset the Merged features.

4.2 Merged datasets

For these experiments, all datasets were merged into one, and the trial-dependent 10-fold cross-validation was used to evaluate the ML models. The results are presented in Figure 2. The results show that the models that use the R-R intervals as input, consistently outperform the models that use GSR features as input.



5. CONCLUSION AND DISCUSSION

We presented an inter-domain study for arousal recognition on six different datasets, recorded with twelve different hardware sensors. We experimented with dataset specific models and models build on the overall (merged) data. We compared the results of seven different ML algorithms, using three different feature inputs (R-R, GSR or Merged – M features).

Table 2. Dataset specific experimental results. Mean accuracy ± stdDev for trial-specific 10-fold cross validation. The best performing models per dataset are marked with green.

Algorithm	Features	Dataset					
		Ascertain	Deap	D. Workload	Cog. Load	Mahnob	Amigos
RF	R-R	0.655 ± 0.07	0.556 ± 0.03	0.785 ± 0.24	0.739 ± 0.13	0.580 ± 0.11	0.536 ± 0.06
	GSR	0.638 ± 0.06	0.503 ± 0.04	0.780 ± 0.24	0.763 ± 0.12	0.611 ± 0.07	0.473 ± 0.11
	M	0.653 ± 0.05	0.540 ± 0.04	0.785 ± 0.25	0.755 ± 0.13	0.611 ± 0.10	0.559 ± 0.10
SVM	R-R	0.664 ± 0.07	0.536 ± 0.05	0.795 ± 0.26	0.717 ± 0.21	0.623 ± 0.15	0.521 ± 0.24
	GSR	0.664 ± 0.07	0.525 ± 0.05	0.795 ± 0.26	0.712 ± 0.20	0.588 ± 0.10	0.470 ± 0.12
	M	0.664 ± 0.07	0.513 ± 0.03	0.795 ± 0.26	0.691 ± 0.18	0.623 ± 0.15	0.506 ± 0.13
GB	R-R	0.649 ± 0.07	0.554 ± 0.03	0.785 ± 0.20	0.736 ± 0.15	0.578 ± 0.11	0.543 ± 0.06
	GSR	0.642 ± 0.05	0.500 ± 0.04	0.800 ± 0.21	0.743 ± 0.12	0.609 ± 0.08	0.527 ± 0.09
	M	0.644 ± 0.05	0.533 ± 0.03	0.755 ± 0.23	0.761 ± 0.15	0.609 ± 0.11	0.542 ± 0.09
AdaB	R-R	0.658 ± 0.06	0.532 ± 0.02	0.750 ± 0.23	0.718 ± 0.13	0.580 ± 0.09	0.531 ± 0.07
	GSR	0.633 ± 0.05	0.485 ± 0.03	0.750 ± 0.22	0.740 ± 0.13	0.589 ± 0.08	0.514 ± 0.09
	M	0.623 ± 0.05	0.526 ± 0.03	0.755 ± 0.22	0.766 ± 0.16	0.610 ± 0.08	0.560 ± 0.08
KNN	R-R	0.625 ± 0.05	0.509 ± 0.02	0.710 ± 0.19	0.715 ± 0.13	0.582 ± 0.07	0.509 ± 0.05
	GSR	0.590 ± 0.06	0.496 ± 0.04	0.795 ± 0.26	0.772 ± 0.09	0.605 ± 0.06	0.533 ± 0.08
	M	0.600 ± 0.05	0.490 ± 0.02	0.750 ± 0.23	0.770 ± 0.13	0.601 ± 0.09	0.533 ± 0.06
NB	R-R	0.654 ± 0.07	0.537 ± 0.04	0.735 ± 0.15	0.748 ± 0.15	0.574 ± 0.06	0.485 ± 0.09
	GSR	0.602 ± 0.04	0.537 ± 0.05	0.540 ± 0.22	0.803 ± 0.09	0.624 ± 0.07	0.454 ± 0.10
	M	0.591 ± 0.04	0.535 ± 0.06	0.665 ± 0.17	0.804 ± 0.12	0.592 ± 0.06	0.486 ± 0.09
DT	R-R	0.664 ± 0.07	0.519 ± 0.05	0.685 ± 0.17	0.736 ± 0.15	0.597 ± 0.09	0.505 ± 0.06
	GSR	0.640 ± 0.05	0.542 ± 0.05	0.765 ± 0.22	0.734 ± 0.08	0.583 ± 0.09	0.483 ± 0.11
	M	0.650 ± 0.05	0.524 ± 0.04	0.615 ± 0.22	0.704 ± 0.09	0.581 ± 0.13	0.551 ± 0.09

The results on the dataset specific setup showed that, out of the ML algorithms tested, none yields the best performance on all datasets. In addition to that, a clear conclusion cannot be made whether the R-R, GSR or the Merged features yield the best results – this is domain (dataset) dependent.

On the merged dataset experiments, the R-R models slightly outperformed the GSR models. This might be due to: (i) having more R-R features that GSR; (ii) having R-R features in frequency domain but no GSR features in frequency domain; (iii) the method for merging the data from the heart-related sensors providing more consistent features across datasets due to less noise in the ECG, BVP data.

In future, we plan to investigate intelligent combinations of ML models in order to gain accuracy. In addition to that, we plan to investigate more advanced techniques such as deep neural networks and transfer learning, which might be able to learn general models that will be able to generalize across different domains. Finally, once we find the best performing scenario, we will generalize the method for arousal recognition to method for valence recognition and method for discrete emotion recognition.

6. REFERENCES

1. R. Picard. Affective Computing. Cambridge, MA: MIT Press, 1997.
2. Depression cost: http://ec.europa.eu/health//sites/health/files/mental_health/docs/matrix_economic_analysis_mh_promotion_en.pdf, [Accessed 27.03.2017].
3. J. A. Russell. A circumplex model of affect. Journal of Personality and Social Psychology, 1980.
4. R. Subramanian, J. Wache, M. Abadi, R. Vieriu, S. Winkler, N Sebe. ASCERTAIN: Emotion and Personality Recognition using Commercial Sensors. IEEE Transactions on Affective Computing. 2016.
5. S. Koelstra, C. Muehl, M. Soleymani, J.-S. Lee, A. Yazdani, T. Ebrahimi, T. Pun, A. Nijholt, I. Patras. DEAP: A Database for Emotion Analysis using Physiological Signals (PDF). IEEE Transaction on Affective Computing, 2012.
6. M.K. Abadi, R. Subramanian, S. M. Kia, P. Avesani, I. Patras. Nicu Sebe. DECAF: MEG-Based Multimodal Database for Decoding Affective Physiological Responses. IEEE Transactions on Affective Computing, 2015.
7. N.R. Lomb. Least-squares frequency analysis of unequally spaced data. Astrophysics and Space Science, vol 39, pp. 447-462, 1976
8. I. Abdic, L. Fridman, D. McDuff, E. Marchi, B. Reimer, Schuller, B. Driver Frustration Detection From Audio and Video. Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'16), 2016.
9. M. Garbarino, M. Lai, D. Bender, R. W. Picard, S. Tognett. Empatica E3 - A wearable wireless multi-sensor device for real-time computerized biofeedback and data acquisition. 4th International Conference on Wireless Mobile Communication and Healthcare, pp. 3-6, 2014.
10. Microsoft band. https://www.microsoft.com/microsoft-band/en-us
11. D. Iacovielloa, A. Petraccab, M. Spezialettib, G. Placidib. A real-time classification algorithm for EEG-based BCI driven by self-induced emotions. Computer Methods and Programs in Biomedicine, 2015.
12. M. Khezria, M.Firoozabadib, A. R. Sharafata. Reliable emotion recognition system based on dynamic adaptive fusion of forehead biopotentials and physiological signals.
13. G. K. Verma, U. S. Tiwary. Multimodal fusion framework: A multiresolution approach for emotion classification and recognition from physiological signals. NeuroImage, 2014.
14. R. M. Mehmooda, H. J. Leea. A novel feature extraction method based on late positive potential for emotion recognition in human brain signal patterns. Computers & Electrical Engineering, 2016.
15. A. Mikuckas, I. Mikuckiene, A. Venckauskas, E. Kazanavicius2, R. Lukas2, I. Plauska. Emotion Recognition in Human Computer Interaction Systems. Elektronika Ir Elektrotechnika, Reserarch Journal, Kaunas University of Technology, 2014.
16. Wei Liu, Wei-Long Zheng, Bao-Liang Lu. Multimodal Emotion Recognition Using Multimodal Deep Learning. Online. Available at: https://arxiv.org/abs/1602.08225, 2016.
17. W-L. Zheng, B-L Lu. A multimodal approach to estimating vigilance using EEG and forehead EOG. Journal of Neural Engineering, 2017.
18. Z. Yin, M. Zhao, Y. Wang, J. Yang, J. Zhang. Recognition of emotions using multimodal physiological signals and an ensemble deep learning model. Comput Methods Programs Biomed. 2017.
19. K.Weiss, T. M. Khoshgoftaar, D. Wang. A survey of transfer learning. Journal of Big Data, 2016.
20. S. Schneegass, B. Pfleging, N. Broy, A. Schmidt, Frederik Heinrich. A Data Set of Real World Driving to Assess Driver Workload. 5th International Conference on Automotive User Interfaces and Interactive Vehicular Applications, 2013.
21. M. Gjoreski, M. Luštrek, M. Gams, H. Gjoreski. Monitoring stress with a wrist device using context. Journal of Biomedical Informatics, 2017, in press.
22. M. Gjoreski, H. Gjoreski, M. Luštrek, M. Gams. Continuous stress detection using a wrist device: in laboratory and real life. ACM Conf. on Ubiquitous Computing, Workshop on mentalhealth, pp. 1185-1193, 2016.
23. M. Soleymani, T.Pun. A Multimodal Database for Affect Recognition and Implicit Tagging, IEEE Transactions On Affective Computing, 2012.
24. L. H. Negri. Peak detection algorithm. Python Implementation. Online. Available at: http://pythonhosted.org/PeakUtils/.
25. M. Wu, PhD thesis. Michigan State University; 2006. Trimmed and Winsorized Eestimators.
26. J.D. Scargle. Studies in astronomical time series analysis. II - Statistical aspects of spectral analysis of unevenly spaced data. The Astrophysical Journal, vol 263, pp. 835-853, 1982.
27. D. P. Kingma, J. Ba. Adam: A Method for Stochastic Optimization, http://arxiv.org/abs/1412.6980, 2014.
28. Tensorflow. Online. Available at: https://www.tensorflow.org/
29. R. Castaldoa, P. Melillob, U. Bracalec, M. Casertaa,c, M. Triassic, L. Pecchiaa. Acute mental stress assessment via short term HRV analysis in healthy adults: A systematic review with meta-analysis. Biomedical Signal Processing and Control. 2015.
30. Scikit-learn, Python machine-learning library http://scikit-learn.org/dev/_downloads/scikit-learn-docs.pdf
31. L.J.P, van der Maaten., G.E. Hinton. Visualizing High-Dimensional Data Using t-SNE. Journal of Machine Learning Research. 9: 2579–2605, 2008.

Predicting Office's Ambient Parameters

Vito Janko
Jožef Stefan Institute
Jožef Stefan International
Postgraduate School
Jamova 39, 1000 Ljubljana
vito.janko@ijs.si

Andreas R. Stensbye
Jožef Stefan Institute
Jamova 39, 1000 Ljubljana
andreas.stensbye@hotmail.com

Mitja Luštrek
Jožef Stefan Institute
Jožef Stefan International
Postgraduate School
Jamova 39, 1000 Ljubljana
mitja.lustrek@ijs.si

ABSTRACT

Bad environmental conditions in the office can negatively affect the workplace productivity. In the presented work we measure three ambient parameters - CO_2, temperature and humidity - asses their quality and predict their likely future values. To do so, we first heuristically determine the state of the office (are the windows open, air conditioner active etc.) and then try to mathematically model the parameter's future behavior. Based on the current and predicted state of ambient parameters, we can send a recommendation on how to best improve them. Experimental evaluation shows that our models outperform the related work in terms of prediction accuracy.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

Keywords

CO_2, temperature, humidity, modeling, recommendations

1. INTRODUCTION

Good work environment is essential for keeping work productivity. In this paper, we are focusing on three office's ambient parameters: CO_2, temperature and humidity. The quality of these parameters is often hard for humans to objectively detect, especially if they are changing slowly. However, it has been shown [4, 5] that when their quality drops below certain thresholds, the work productivity in the office is negatively affected.

In this paper we present an intelligent system that is able to measure these parameters and estimate their future values. In the case of CO_2 and temperature, a simple mathematical model is used for prediction, in the case of humidity a machine learning model is used instead. Furthermore, it is able to asses the quality of these parameters, simulate several possible actions that an user can take, and then recommend the one leading to the best working conditions. The system is meant to be used in offices without automatic ambient control and is a part of the larger "Fit4Work" project [2] that is focused on helping to raise the well-being of office workers. It requires no prior knowledge or manual input of office properties, yet it is able to adapt to them over time.

The ambient parameters were measured using the Netatmo commercial device [1]. The same device is expected to be used by the end users of the system, although it can be replaced with any similar device with the same functionality. This device has an indoor and an outdoor unit, both capable of measuring the CO_2, temperature and humidity, and sending the data to a web server. For easier testing and validation of our method we also had sensors that monitor if the windows are opened and closed and an application where we manually labeled the number of people in the office, the air conditioner state, heater state and humidifier state. As for the time of writing this paper we collected roughly two years of data for three different offices in our department. Data is continuously sent to a web server, where it is analyzed as described in Section 2. If a recommendation is deemed necessary, it is sent to a mobile device via a push notification.

The paper was inspired by another work [3], and proposes a different solution to the same problem. The proposed solution makes heavier use of mathematical modeling and produces more accurate predictions about the ambient parameter's future values.

2. METHODOLOGY

The goals of this paper are three-fold. First, to predict the state of the office: are the windows open, is air conditioner turned on, etc. This not only allows us to predict the probable changes in the ambient parameters, but also to make sensible recommendations: no need to recommend opening of the windows if they are already open. Second, to predict the future behavior of the following three parameters - CO_2, temperature and humidity. We are interested in predicting values up to 30 minutes in advance. Our data was measured every 10 minutes, so this corresponds to 3 data points. Behavior should be predicted for the current state and also for the cases where some office parameters changes. Finally, we use a combination of the previous two points to form recommendations to the user on actions that improve the work environment.

While the physical phenomena of temperature, humidity and CO_2 was already heavily studied in the past, the challenge we face here is in not knowing any attributes of the office where this system would be used: how big the office, how good the thermal insulation, the surface of the windows, etc. Using standard formulas for predicting the ambient parameters can therefore be infeasible, given how many unknowns they contain. In our approach we tried to simplify the models to simple versions with only a few unknowns. We use data recorded in the target office in the last two weeks (exact number of days may vary based on office usage) to estimate these unknowns, and then use them for real-time predictions in the following day.

2.1 Virtual sensors

Virtual sensors refer to values that are not directly measured. Instead, their value is derived from the measured data and then later used to help derive some other value. In our setting, there are five virtual sensors that affect the ambient parameters: the windows state, air conditioner state, heater state, humidifier state and number of people in the office.

”The number of people in the office” is calculated from raising CO_2 levels, and the humidifier state is tied to humidity data, so those two will be explored in the corresponding Sections 2.2 and 2.4. The remaining three can be determined with simple heuristics as described below.

2.1.1 Windows

Windows were modeled in a binary fashion: they are either open or they are closed. In a real office there might be many windows, some of them open, some closed, some perhaps half-open at any time; but lacking any knowledge about the window quantity or size, predicting their state more accurately is almost impossible.

An effect of opening the window is reflected on all three ambient parameters, but only in the case of CO_2 is the effect consistent. Whenever a window is opened, CO_2 falls drastically, whenever it is closed it starts to rise again. This allows us to make a simple heuristic: a.) if the CO_2 is falling faster then some threshold, window was opened; b.) if CO_2 keeps increasing, the window is closed; c.) if neither of those is happening, assume the last known state. Thresholds can be determined by looking at the data history and find such values that would generate predictions, where windows is opened/closed few times a day, as would realistically be the case.

This approach could be improved by correlating changes in CO_2 to those in temperature and humidity, but the described simple heuristic appeared to work well in practice.

2.1.2 Air conditioner

Again we assume binary outcome - the air conditioner is either on or off - additionally we assume that the temperature set on it is constant, or at least is changing infrequently. The distinguishing pattern of air conditioning is one of temperature inside decreasing while the temperature outside is higher then inside. Since the temperature naturally tries to equalize itself with its surroundings and since all other factors (people, computers, etc..) only serve to warm the office, it is reasonable to conclude that such a temperature drop was caused by the air conditioner. After a while of the air conditioner working, the temperature will converge to value that can be stored for later predictions. If the temperature starts rising again, the air conditioner is assumed to be turned off.

2.1.3 Heater

The same assumptions and methods are used here as with the air conditioner, except in reverse: the heater is on if the inside temperature rises significantly more then expected from the outside temperature, etc.

2.2 CO_2 predictions

We start by modeling CO_2, as it the most ”well-behaved” of the three ambient parameters, and we describe the process in depth. We later use a similar methodology for temperature modeling. Intuitively, CO_2 level inside the office is increasing linearly with respect to the number of people present, but at the same time it tries to equalize itself with the outside CO_2 level. The bigger the difference between outside and inside, the faster it moves from one side to another. If window is opened, the same happens, only to a significantly larger degree. This can be encapsulated in the following equation.

$$C_{n+1} = C_n + \alpha(C_{out} - C_n) + \beta p \qquad (1)$$

$C_n = CO_2$ inside at timestep n
$C_{out} = CO_2$ outside
$p =$ the number of people in the room
$\alpha =$ the coefficient of diffusion speed (between 0 and 1) - small for closed windows, big for open ones
$\beta =$ how much a single person raises CO_2 in a given time unit

Using all the labeled data, the α and β are mostly trivial to compute using linear regression. Using them results in an almost perfect match between the predicted and real values. In Figure 1 we plot a scenario where we know the initial CO_2 level and all future windows states and all future numbers of people, and we are able to predict CO_2 level two days in advance. This strongly signifies that the model captures the real-life behavior of CO_2, and it is only a matter of determining the correct coefficients.

Calculating the coefficients for a given office without the labeled data, however, is a challenging task as the above formula has 5 unknowns - α when windows are closed, α when windows are opened, window state, β and number of people p. Furthermore these coefficients can behave very similarly: CO_2 level in a room with many people and open window can be close to CO_2 level in a room with closed windows and few people. The first improvement is to combine the two variables β and p into one - γ, as we never need those two individually and are only interested in their product. This shortens the formula to:

$$C_{n+1} = C_n + \alpha(C_{out} - C_n) + \gamma \qquad (2)$$

This formula can be rewritten in an analytical way (Equation 3) so it can predict an arbitrary time step instead of only steps of integer size (10 minutes). A simple explanation of this formula goes as follows - CO_2 always converges to a value L. The number of people in the office dictates this limit, while the value α dictates how fast we approach this limit. The inverse of this formula will also be useful and can be trivially computed using some basic algebra.

$$C_n = \begin{cases} \gamma^n, & \text{if } \alpha = 0 \\ L + (C_0 - L)(1-\alpha)^n, & \text{otherwise} \end{cases} \tag{3}$$

$$L = C_{out} + \frac{\gamma}{\alpha}$$

Determining the window state is described in Section 2.1.1. If we know the α value for the current window state, the γ value becomes the only unknown in the formula and can be determined with a simple linear regression, using last three data points. Since γ correlates with the number of people in the office, it must be recalculated for every prediction. The α value on the other hand is dependent on the office heat insulation level, office size and windows size, and is therefore a constant. We can therefore estimate the α value by trying different values on the past two weeks of data and then select the one that has the lowest error rate when predicting - this is possible since when predicting on the past data, we already know what CO_2 value will be reached.

2.3 Temperature predictions

We used the same base formula - Equation 3 - for the inside temperature prediction. This model however, has to be made more complex because of two factors.

First, the temperature does not converge towards the outside one, but goes towards some function of the outside temperature instead. For example, even if the outside temperature is below zero, the temperature in the office never went below 10 degrees, even without heating. There are several reasons for this behavior, including the heat of the building itself, and the fact that building is warming and cooling at different rates than the exterior when the external temperature changes. This is dealt by calculating a function from last two weeks of data that models the expected inside temperature as a linear function of the outside temperature. The calculation is made during rest days, when no one is in the office, reducing the noise in the data. This calculated value then replaces the value C_{out} in the Equation 3.

Second, we have to account for both air conditioning and heating. The detection of their state is described in Sections 2.1.2 and 2.1.3. In the same section it is also described how to collect the limiting temperature value these devices generate. If either device is on, the corresponding limiting value replaces L in Equation 3. Improvement of this rather simplistic modeling of the devices is subject to future work. A prediction example is plotted in Figure 2.

2.4 Humidity predictions

Humidity was not changing much in our data, and when it did, there was no obvious pattern. So instead of plugging the data into the same equation, we used a classical machine learning approach. The last few humidity and temperature measurements, together with the window state and estimated number of people (computed from γ in CO_2 model) are fed into a machine learning model, and a prediction for future humidity is given. Again the training of the model is made on the previous two weeks. If it turns out that the prediction underestimated the humidity in the office, the humidifier is determined to be active. If the classifier overestimates the humidity and humidifier was considered active, it is considered inactive from then on.

2.5 Recommendation system

Each ambient parameter has predefined quality ranges - good, medium and bad. For example: ”good” CO_2 is under 500 ppm, ”bad” over 800 ppm and ”medium” in between. The ideal case is to have all three parameters in the ”good” quality range. This, however, is not always possible as improving one parameter may damage another - opening the window may improve the CO_2, but it may reduce the temperature quality. The priority of the system is to have the minimum number of ”bad” parameters. If all the parameters are ”medium” or above, the maximum number of ”good” parameters is prioritized.

A possible action is a change in one of the devices/windows that exist in office. In the current version, all the devices are assumed to be binary (air conditioner is either on or off, windows are opened or closed, etc.). The list of all possible actions is generated based on the current assumed state of the office. If the windows are assumed opened, ”open the window” action will be omitted. Some hand-selected actions may appear in pairs, as they are commonly done simultaneously: turn on the air conditioner and close the windows for example. A default action ”do nothing” is also included on the list.

Each action effect is simulated over the period of 30 minutes. The action that results in the best state after that time interval is selected. If the action has a higher score than the default action of doing nothing, it is recommended to the user.

While not fully implemented yet, there are two areas with possible improvements that are currently worked on. One is to try to make the recommendations more time-specific. Instead of ”open the window”, we could recommend ”open the window for 7 minutes, then close again”. This can be done by first determining all the relevant time frames - times where a parameter shifts from one quality range to another. All the possible actions can then be tested against every relevant time frame. Second is to predetermine which actions are even sensible, given the context. If the only problem is the temperature inside being too cold and it is also cold outside, then the sensible options are only to close the window or to turn on the heater. This is being implemented by an ontology that contains facts about some ambient parameters, configured in a way that is able to search for relevant actions given current state.

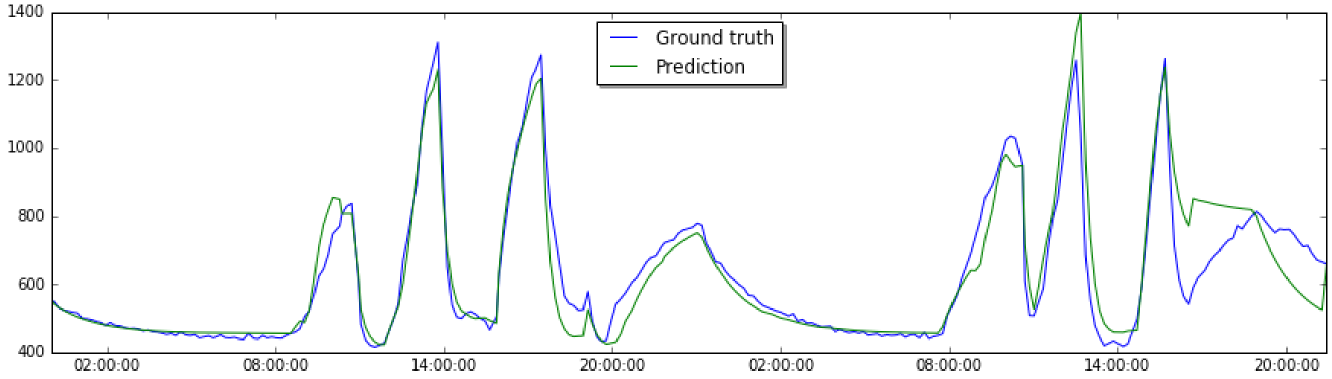


Figure 1: CO$_2$ prediction and ground truth, predicting values for the next two days, supposing that we have perfect information about the current and future office state.

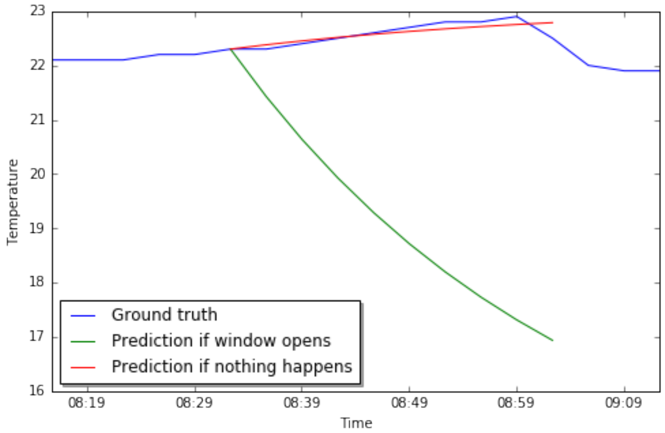


Figure 2: Temperature prediction and ground truth. We predict what happens if no action is done, against what happens if window is opened. The prediction starts in the past so we can compare it to the actual measurements.

3. RESULTS

As results we list (Table 1) the mean absolute error when predicting a parameter 30 minutes in advance, during a three month period. The test are made to be comparable with those in paper by Frešer et al. [3]. We show that our predictions for CO$_2$ and temperature display lower error then the before-mentioned work. Their humidity measurements were better, probably because of better selection of features in their model.

Table 1: Mean absolute error

Parameter	Our error	Error reported by Frešer [3]
CO$_2$ [ppm]	43	79
Temperature [°C]	0.36	0.50
Humidity [%]	1.2	0.74

4. CONCLUSIONS

In this work we model three ambient parameters in the office. For two of them, we show a simple mathematical model that predicts their future behavior. For those two we get more accurate predictions than those in the related work. This is probably a consequence of using a physically-inspired formula. For humidity we use a machine learning model that while showing promising results, still has room for improvement. We also predict the state of devices and windows in the office, although the accuracy of this prediction has not yet been directly tested. Furthermore we presented a recommendation system that we plan to test with multiple real offices in the future.

5. REFERENCES

[1] ”netatmo. 2016. netatmo. (2016)”. ”https://www.netatmo.com”. ”Accessed: 2017-09-03”.

[2] B. Cvetković, M. Gjoreski, M. Frešer, M. Kosiedowski, and M. Luštrek. Monitoring and management of physical, mental and environmental stress at work.

[3] M. Frešer, A. Gradišek, B. Cvetković, and M. Luštrek. An intelligent system to improve thc parameters at the workplace. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, pages 61–64. ACM, 2016.

[4] L. Lan, P. Wargocki, and Z. Lian. Optimal thermal environment improves performance of office work. *REHVA European HVAC Journal*, 2:12–17, 2012.

[5] D. P. Wyon. The effects of indoor air quality on performance and productivity. *Indoor air*, 14(s7):92–101, 2004.

Real-time Content Optimization in Digital Advertisement

Tom Vodopivec
University of Ljubljana
Faculty of Computer and Information Science
tom.vodopivec@fri.uni-lj.si

Davor Sluga
University of Ljubljana
Faculty of Computer and Information Science
davor.sluga@fri.uni-lj.si

Nejc Ilc
University of Ljubljana
Faculty of Computer and Information Science
nejc.ilc@fri.uni-lj.si

Gregor Šušelj
Celtra
Data Insights Team
gregor.suselj@celtra.com

Rok Piltaver
Celtra
Engineering Analytics Team
rok.piltaver@celtra.com

Domen Košir
Celtra
Data Insights Team
domen.kosir@celtra.com

ABSTRACT

A key goal of advertising industry is to present the target audience with advertisements that induce most engagement. In digital advertising we are able to collect huge amounts of data on how different advertisements perform. This data can be used to optimize the content of such advertisements in real-time. The idea behind the optimization is essentially the same as in the multi-armed bandit problem. There are many optimization algorithms available for solving it, but they need to be modified for the specifics of digital advertising. In this study, we analyse real data from hundreds of advertising campaigns and present a methodology to asses the potential of advertisement content optimization. We compare the performance of different optimization algorithms and propose their modifications. We conclude that only a small part of the advertising campaigns can potentially benefit from content optimization. However, when there is room for improving the performance of an advertisement, the optimization algorithms coupled with the proposed modifications are able to exploit most of it.

Keywords

Digital advertising, optimization, multi-armed bandit, selection bias, exploration-exploitation trade-off, Thompson sampling, Upper confidence bounds algorithm

1. INTRODUCTION

In the advertising industry, there is a persistent aspiration to create and present to the target audience the most relevant advertisement, which would produce the highest number of *engagements* from the viewers. The most frequently measure of engagement, specific to the digital advertising, is the *click rate*. It measures how many views of a digital advertisement led to a user clicking on the advertisement and thus showing interest in the advertised product or service.

Given the complexity of the industry and the target population, it is difficult to predict which advertisement or variant of an advertisement would be more engaging. Fig. 1 shows an example of two variants of an advertisement. Digital advertising enables collecting large amounts of data that can be used to measure the performance of advertisements. Using an appropriate comparison methodology one can then decide which advertisement variant should be presented to the target audience. Some research regarding optimization of digital advertisements has been published [5, 7], however a lot remains hidden from the scientific literature due to confidentiality agreements in the advertising industry.

Advertisement content optimization can be done for long time spans – the decision on the content used in the advertisements for the next campaign is made based on the engagement rates in the previous campaign. On the other hand, optimization can also be done in real-time – the decision on the content is made when an advertisement is about to be shown on a web page or in a mobile application. In this paper, we are focusing on the latter approach. Our primary goal is to estimate the *potential* of the real-time content optimization, i.e. the differences in engagement rates among different variants of an advertisement. If there is some potential present, then the goal is to find the best algorithms that optimize the content of advertisements in real-time.

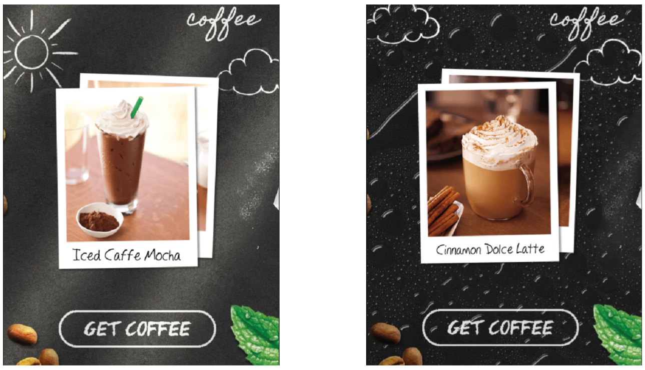
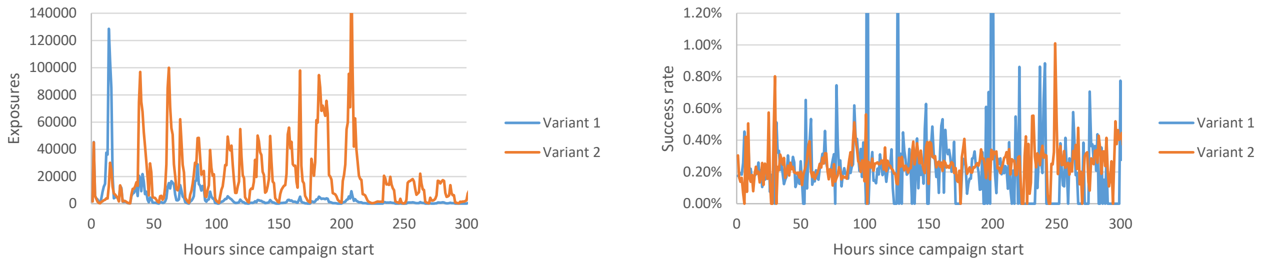


Figure 1: Which advertisement is more engaging?

2. CHALLENGES IN ADVERTISEMENT CONTENT OPTIMIZATION

A digital advertising *campaign* usually consists of different *variants* of an advertisement, and generally multiple possible *media* (different web pages and mobile applications) where an advertisement is displayed. The variants of an advertisement can differ in many ways (e.g. text, graphics, colour). As the campaign progresses, one can track the num-



(a)

(b)

Figure 2: Sample time lapse of exposures (a) and success rates (b) for an advertisement campaign with two variants.

ber of *exposures* (i.e., how many times it was displayed to a viewer) and the number of engagements or *successes* (i.e., the number of clicks on the advertisement) of all variants on a time basis (we collected hourly data). Our hypothesis is that this data can be used to identify the better performing variants and adjust the amount of times they are displayed in their favour. In practice, this is not a trivial task. The number of successes is usually very low in comparison to the number of exposures, yielding near-zero probabilities of success. Additionally, the data is non-stationary as shown on an example campaign in Fig. 2: the volume of exposures and the probability of success is changing over time, because of hourly fluctuations in advertisement traffic, audience variability, and seasonal trends. Therefore, large amounts of data are needed to obtain reliable estimates of each variants' performance. Obtaining additional data is usually not feasible, since exposures cost money and there are limited funds available for a campaign. Another problem is that any optimization done during the campaign introduces a *selection bias* [12] into the data – the variants chosen by the optimization algorithm have positively-skewed success rates. As a consequence, performance of certain variants can not be reliably estimated due to low number of exposures.

When optimizing the advertisement content, one has to carefully set the trade-off between the amount of exposures available to the optimization algorithm and the amount of exposures used for performance analysis. The more aggressive the optimization, the more biased (less reliable) are the estimates of the variants' performance. The balance of exposures across different variants must be such that the optimization algorithm will yield the highest possible number of engagements, while still allowing estimation of the performance gain with a reasonable accuracy. Another obstacle, caused by technical limitations, is that feedback on how variants perform is usually delayed (in our case by a few hours) and computed in batches (e.g., hourly). Therefore, the optimizing algorithm has to set the distribution of the exposures across different variants on a hourly basis and not for each exposure separately – this has a negative effect on the optimization potential of the algorithm.

3. EVALUATING THE PERFORMANCE

One approach for evaluating the performance of a digital advertisement content optimization algorithm is to compute its *lift* $L = \frac{S_o}{S_r} - 1$, where S_o is the number of successes achieved by the optimization algorithm and S_r is the number of successes achieved by displaying the variants randomly according to the uniform distribution. We usually express lift in percentages. A positive lift means that using the optimization algorithm is better than choosing variants randomly, whereas a negative lift means that the optimization has a negative effect on the success rate. We estimate the *potential* of a campaign by computing the lift for the *oracle* algorithm, which is the optimal algorithm that always (in each hour) chooses the best performing variant. Of course, the oracle algorithm can only be used on the past data (not in real-time), when it already knows how variants performed.

To cope with the large variance of the success rates, we use one real campaign to generate several artificial campaigns by applying different amounts of smoothing to compute hourly success rates for each variant. Only the uniformly selected variants are used in creating the artificial campaigns to avoid selection bias affecting the success rates. Smoothing ranges from producing completely stationary data on one hand to highly volatile trends (no smoothing) on the other hand. From these campaigns we choose the best-case and the worst-case campaign according to its potential. This gives us the upper and lower limit for the true potential for different scenarios, which could be hidden in the real data. The artificial campaigns also enable us to perform an arbitrary number of runs of the optimisation algorithms in order to evaluate their performance. This would not be possible on real campaigns, since it is very expensive.

To understand the reliability of the measured lift, it is crucial to compute its confidence bounds. These can be computed using the *Fieller's theorem* [6] for the confidence interval of the ratio of two means. To avoid the selection bias, we split the exposures into two sets when running the campaign: the *control set*, where the variants are displayed at random, and the *optimization set*, where variants are displayed according to the decisions of the optimization algorithm. The control set is used to estimate the success rates of each variant and deduce the performance of the optimization algorithm. The control set is usually small ($\sim$ 10% of exposures), so the optimization algorithm has a big enough volume of exposures left to optimize. Low number of exposures in the control set has the disadvantage of confidence intervals being rather wide and the lower bound of the lift often being negative. We improve the estimation of the bounds and still keep the same amount of exposures selected by the algorithm by introducing *three-set sampling* [12]. Here, we split the optimization set into a *learning set* and *evaluation set.* The control set is composed of the exposures with randomly-selected variants. The learning set is composed of the exposures that were selected by the optimization algorithm and are used by it for further selection. The evaluation set is composed of the exposures that were selected by the algorithm, but are not used by it for further selection – the optimization does not consider the exposures in the evaluation set when deciding which variant to chose next. Data from the control set and the evaluation set can then be used to produce unbiased estimates of the lift and its confidence bounds.

4. OPTIMIZATION ALGORITHMS AND THEIR IMPROVEMENTS

The optimization of advertisement variant selection can be treated as the *multi-armed bandit* problem [10]. This is a problem in which a gambler at a row of slot machines (one-armed bandits) has to decide which machines to play, how many times to play each machine and in which order to play them. Each machine provides a random reward from a probability distribution specific to that machine. The goal of the gambler is to maximize the sum of rewards earned through a sequence of plays. A plethora of algorithms dedicated to solving it exist [4], ranging from simple ones like the *ϵ-greedy* selection policy, to more intricate ones like the *Thompson sampling (TS)* [1], the *upper confidence bound algorithm (UCB) [2]*, and its improvements [3]. Each optimization algorithm balances the *exploration-exploitation trade-off* using its parameters. They instruct the algorithm what will be the ratio between the number of exposures it will use to display the percievingly best variant (i.e., exploitation) and the number of exposures it will use to display the other variants to see if some other variant is potentially better (i.e., exploration).

By default, the aforementioned algorithms are not intended to cope with delayed feedback, batch updates, and non-stationary data. Therefore, we propose several enhancements to improve their performance on our use-case. A solution for the delayed feedback and batch updates is to simulate the immediate feedback for each variant selection based on the historical success rates. When the real feedback becomes available all of the simulated exposures are discarded and replaced by the real data.

Non-stationarity of the data is problematic because the algorithms have trouble adapting to trend changes. We deal with abrupt changes in trends by using the *Page-Hinkley test* [8] for detecting change-points in the success rate and the number of exposure displayed per time unit. If the test detects a change, we discard a portion of historical data to enable faster adaptation of the optimization algorithm to the new situation. Another approach that we propose is to perform periodic *forgetting* [10] of historical data based on some predefined condition like the total number of exposures or time elapsed since the last forgetting event. After the condition is met a forgetting event is triggered, which causes a part of the historical data to be discarded.

The above two approaches are suitable if there are a lot of abrupt changes in the trends. However, if the changes happen more gradually a more conservative approach to forgetting of historical data is needed. To deal with gradual changes in the trends we implement a *two-memory structure* [9] of the historical data. We keep track of exposures and successes in two separate data structures: a *long-term (persistent) memory* and *short-term (transient)* memory. The first holds almost all historical data and the latter holds only very recent data about each variant. A weighted sum of the two is then used to estimate the success rates of the variants and feed them into the optimization algorithm.

5. DATA AND EXPERIMENTS

We collected data from hundreds of advertising campaigns that used content optimization. The data includes the number of exposures and successes for each variant for each hour of each campaign. Exposures in each campaign were split in two groups, for 10% of the exposures a random advertisement variant was chosen from an uniform distribution (i.e., the control dataset), whereas for the remaining 90% of the exposures a hand-tuned optimization algorithm chose the variant.

From all the campaigns, we selected a representative sample of 38 campaigns for further analysis. The durations of the selected campaigns span from 3 days to 6 months and the number of exposures ranges between $10,000$ and $10,000,000$. There are 2 to 10 advertisement variants per campaign, with success rates ranging from 0.1% to 20% (median is 1.8%). The trend-change analysis shows there are 2 to 6 events per campaign that change the success rates of the variants, apart from the monthly, weekly, and daily seasonalities.

A preliminary analysis showed that 20 campaigns (out of 38) have at least some potential, so we used these to generate two sets of 200 artificial campaigns: a best-case and worst-case set. We created 10 artificial campaigns from each real campaign (hence 200 from 20) per set. The two artificial sets were then used to estimate the lower and upper bound of the real campaigns' potential and to measure the performance of the optimization algorithms. We tested three optimization algorithms: ε-greedy, Thompson sampling, and UCB1-Tuned. Each algorithm was run 10 times on each artificial campaign, producing 2000 runs per artificial set. Based on these 2000 runs per algorithm we comparatively analysed how many times each algorithm achieved lifts above 5%, 10%, and 20%. The simulations on artificial campaigns allocated 10% of exposures to the control set, 45% to the learning set, and 45% to the evaluation set.

6. RESULTS AND DISCUSSION

We observe that approximately half of the campaigns have no significant potential (Table 1): in the best case there is 45% of campaigns with lift below 5% and in the worst case there is 74% of such campaigns. A deeper analysis of the advertisement variants showed that there are considerable differences between the variants in some campaigns, whereas the variants are nearly identical in other campaigns, which reduces the amount of optimization potential.

In the initial experiments, we observed that ε-greedy is too aggressive for our non-stationary problem – it quickly fits to a single variant and requires a long time to switch to another. Therefore, we omit it from further experiments

Table 1: The theoretical potential in optimizing advertisement campaigns.

Ratio [%] of real campaigns with potential lift above:	5%	10%	20%
Worst case	26	18	8
Best case	55	53	34

Table 2: The performance of optimization algorithms. We present the lower and upper bounds based on the results from the worst-case and best-case sets of artificial campaigns, respectively.

Ratio [%] of artificial campaigns with opt. lift above:	5%	10%	20%
Thompson sampl.	25–39	17–27	11–16
UCB1-Tuned	30–44	24–35	13–19
Imp. Thompson sampl.	36–55	23–37	12–30
Imp. UCB1-Tuned	34–59	27–35	14–30

and focus on the other two algorithms instead. Comparison of the original Thompson sampling and UCB1-Tuned algorithms shows the latter is better (Table 2). When the two algorithms are improved with the techniques presented in Section 4, both Thompson sampling and UCB1-Tuned perform almost equally. The (improved) optimization algorithms exploit approximately 25% to 50% of the potential in an advertisement campaign. Note that exploiting 100% of the potential is not possible.

We omit the in-depth analysis of the proposed improvements of the optimization algorithms, and provide just a brief summary of the findings. We observed that simulated immediate feedback only slightly improves performance, that two-memory architecture is significantly beneficial, and that periodic forgetting (with two memories) is as good as trend-change detection with the Page-Hinkley statistical test. We have not measured which forgetting approach is more resistant to parameter over-fitting. When the algorithms are improved with the techniques mentioned above, the exploratory parameters of the optimization algorithms can be adjusted to make the algorithms more aggressive, since forgetting resets them frequently enough so they don't stick to the same variant for too long.

7. CONCLUSIONS

In this study, we propose a methodology for estimating the potential behind real-time optimization of digital advertisement content. We analysed hundreds of advertising campaigns provided by Celtra and developed a methodology for generating artificial campaigns that mimic real campaigns. We generated multiple benchmark sets of artificial campaigns and used them to empirically evaluate several optimization algorithms in combination with different improvements. The proposed modifications proved highly beneficial and provided us with ideas that may increase the performance of the optimization algorithms even further. Our main discovery is that optimization algorithms are able to exploit the potential of an advertising campaign reasonably well. However, many campaigns have no potential at all – campaigns with no significant differences between variants do not benefit from optimization. In future work, we would like to identify what types of campaigns exhibit high potential. Discovering the characteristics that make a campaign suitable for optimization in the first place may significantly increase the value of digital content optimization in digital advertising. Additionally, more advanced algorithms, like dual-layer UCB or budget-limited UCB [11], could be used to further increase the benefit of optimization. Another possible improvement would be to find the optimal relative sizes of sets in the three-set sampling we used because large control and evaluation sets provide more data for the analysis, but inhibit the algorithms' learning rate and hence performance.

8. ACKNOWLEDGMENTS

We would like to thank Grega Kešpret and Gregor Smrekar for the valuable insight they provided during our study.

9. REFERENCES

[1] S. Agrawal and N. Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on Learning Theory*, pages 39–1, 2012.

[2] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time Analysis of the Multiarmed Bandit Problem. *Machine Learning*, 47(2-3):235–256, 2002.

[3] P. Auer and R. Ortner. UCB revisited: Improved regret bounds for the stochastic multi-armed bandit problem. *Periodica Mathematica Hungarica*, 61(1):55–65, 2010.

[4] V. Kuleshov and D. Precup. Algorithms for multi-armed bandit problems. *Journal of Machine Learning*, 1:1–32, 2014.

[5] T. Lu, D. Pál, and M. Pál. Contextual multi-armed bandits. In *Proceedings of the Thirteenth international conference on Artificial Intelligence and Statistics*, pages 485–492, 2010.

[6] R. Moineddin, J. Beyene, and E. Boyle. On the location quotient confidence interval. *Geographical Analysis*, 35(3):249–256, 2003.

[7] S. Pandey and C. Olston. Handling advertisements of unknown quality in search advertising. In *Advances in neural information processing systems*, pages 1065–1072, 2007.

[8] R. Sebastiao and J. Gama. A study on change detection methods. In *Progress in Artificial Intelligence, EPIA*, pages 12–15, 2009.

[9] D. Silver, R. S. Sutton, and M. Müller. Sample-based learning and search with permanent and transient memories. In *Proceedings of the 25th international conference on Machine learning*, pages 968–975, 2008.

[10] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction.* The MIT Press, 1998.

[11] L. Tran-Thanh, A. C. Chapman, A. Rogers, and N. R. Jennings. Knapsack based optimal policies for budget-limited multi-armed bandits. In *AAAI*, 2012.

[12] M. Xu, T. Qin, and T.-Y. Liu. Estimation bias in multi-armed bandit algorithms for search advertising. In *Advances in Neural Information Processing Systems*, pages 2400–2408, 2013.

Indeks avtorjev / Author index

Konferenca / Conference

Uredili / Edited by

Slovenska konferenca o umetni inteligenci / Slovenian Conference on Artificial Intelligence

Matjaž Gams, Mitja Luštrek, Rok Piltaver