

IS
20
19

Zbornik 22. mednarodne multikonference

INFORMACIJSKA DRUŽBA

Zvezek A

Proceedings of the 22nd International Multiconference

INFORMATION SOCIETY

Volume A

Slovenska konferenca o umetni inteligenci

Slovenian Conference on Artificial Intelligence

Uredili / Edited by

Mitja Luštrek, Matjaž Gams, Rok Piltaver

<http://is.ijs.si>

10. oktober 2019 / 10 October 2019

Ljubljana, Slovenia

Zbornik 22. mednarodne multikonference
INFORMACIJSKA DRUŽBA – IS 2019
Zvezek A

Proceedings of the 22nd International Multiconference
INFORMATION SOCIETY – IS 2019
Volume A

Slovenska konferenca o umetni inteligenci
Slovenian Conference on Artificial Intelligence

Uredili / Edited by

Mitja Luštrek, Matjaž Gams, Rok Piltaver

<http://is.ijs.si>

10. oktober 2019 / 10 October 2019
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

Na naslovnici je uporabljena slika robota podjetja 

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

Ljubljana, oktober 2019

Informacijska družba
ISSN 2630-371X

Kataložni zapis o publikaciji (CIP) pripravili v Narodni in univerzitetni knjižnici v Ljubljani COBISS.SI-ID=302443008 ISBN 978-961-264-155-9 (epub) ISBN 978-961-264-156-6 (pdf)

PREDGOVOR MULTIKONFERENCI INFORMACIJSKA DRUŽBA 2019

Multikonferenca Informaci družba (<http://is.ijs.si>) je z dvaindvajseto zaporedno prireditvijo tradicionalni osrednji srednjeevropski dogodek na področju informacijske družbe, računalništva in informatike. Informacijska družba, znanje in umetna inteligenca so - in to čedalje bolj – nosilci razvoja človeške civilizacije. Se bo neverjetna rast nadaljevala in nas ponesla v novo civilizacijsko obdobje? Bosta IKT in zlasti umetna inteligenca omogočila nadaljnji razcvet civilizacije ali pa bodo demografske, družbene, medčloveške in okoljske težave povzročile zadušitev rasti? Čedalje več pokazateljev kaže v oba ekstrema – da prehajamo v naslednje civilizacijsko obdobje, hkrati pa so notranji in zunanji konflikti sodobne družbe čedalje težje obvladljivi.

Letos smo v multikonferenco povezali 12 odličnih neodvisnih konferenc. Zajema okoli 300 predstavitev, povzetkov in referatov v okviru samostojnih konferenc in delavnic in 500 obiskovalcev. Prireditvev bodo spremljale okrogle mize in razprave ter posebni dogodki, kot je svečana podelitev nagrad. Izbrani prispevki bodo izšli tudi v posebni številki revije Informatica (<http://www.informatica.si/>), ki se ponaša z 42-letno tradicijo odlične znanstvene revije.

Multikonferenco Informacijska družba 2019 sestavljajo naslednje samostojne konference:

- 6. študentska računalniška konferenca
- Etika in stroka
- Interakcija človek računalnik v informacijski družbi
- Izkopavanje znanja in podatkovna skladišča
- Kognitivna znanost
- Kognitonika
- Ljudje in okolje
- Mednarodna konferenca o prenosu tehnologij
- Robotika
- Slovenska konferenca o umetni inteligenci
- Srednje-evropska konferenca o uporabnih in teoretičnih računalniških znanostih
- Vzgoja in izobraževanje v informacijski družbi

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

V 2019 bomo sedmič podelili nagrado za življenjske dosežke v čast Donalda Michieja in Alana Turinga. Nagrado Michie-Turing za izjemen življenjski prispevek k razvoju in promociji informacijske družbe je prejel prof. dr. Marjan Mernik. Priznanje za dosežek leta pripada sodelavcem Odseka za inteligentne sisteme Instituta »Jožef Stefan«. Podeljujemo tudi nagradi »informacijska limona« in »informacijska jagoda« za najbolj (ne)uspešne poteze v zvezi z informacijsko družbo. Limono je dobil sistem »E-zdravje«, jagodo pa mobilna aplikacija »Veš, kaj ješ?!«. Čestitke nagrajencem!

Mojca Ciglarič, predsednica programskega odbora

Matjaž Gams, predsednik organizacijskega odbora

FOREWORD - INFORMATION SOCIETY 2019

The Information Society Multiconference (<http://is.ijs.si>) is the traditional Central European event in the field of information society, computer science and informatics for the twenty-second consecutive year. Information society, knowledge and artificial intelligence are - and increasingly so - the central pillars of human civilization. Will the incredible growth continue and take us into a new civilization period? Will ICT, and in particular artificial intelligence, allow civilization to flourish or will demographic, social, and environmental problems stifle growth? More and more indicators point to both extremes - that we are moving into the next civilization period, and at the same time the internal and external conflicts of modern society are becoming increasingly difficult to manage.

The Multiconference is running parallel sessions with 300 presentations of scientific papers at twelve conferences, many round tables, workshops and award ceremonies, and 500 attendees. Selected papers will be published in the *Informatica* journal with its 42-years tradition of excellent research publishing.

The Information Society 2019 Multiconference consists of the following conferences:

- 6. Student Computer Science Research Conference
- Professional Ethics
- Human – Computer Interaction in Information Society
- Data Mining and Data Warehouses
- Cognitive Science
- International Conference on Cognitonics
- People and Environment
- International Conference of Transfer of Technologies – ITTC
- Robotics
- Slovenian Conference on Artificial Intelligence
- Middle-European Conference on Applied Theoretical Computer Science
- Education in Information Society

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

For the fifteenth year, the award for life-long outstanding contributions will be presented in memory of Donald Michie and Alan Turing. The Michie-Turing award was given to Prof. Marjan Mernik for his life-long outstanding contribution to the development and promotion of information society in our country. In addition, a recognition for current achievements was awarded to members of Department of Intelligent Systems of Jožef Stefan Institute. The information lemon goes to the “E-Health” system, and the information strawberry to the mobile application “Veš, kaj ješ?!” (Do you know what you eat?!). Congratulations!

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

KONFERENČNI ODBORI

CONFERENCE COMMITTEES

International Programme Committee

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

Organizing Committee

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

Programme Committee

Mojca Cigliarič, chair
Bojan Orel, co-chair
Franc Solina
Viljan Mahnič
Cene Bavec
Tomaž Kalin
Jozsef Györkös
Tadej Bajd
Jaroslav Berce
Mojca Bernik
Marko Bohanec
Ivan Bratko
Andrej Brodnik
Dušan Caf
Saša Divjak
Tomaž Erjavec
Bogdan Filipič

Andrej Gams
Matjaž Gams
Mitja Luštrek
Marko Grobelnik
Vladislav Rajkovič
Grega Repovš
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
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

KAZALO / TABLE OF CONTENTS

Slovenska konferenca o umetni inteligenci / Slovenian Conference on Artificial Intelligence	1
PREDGOVOR / FOREWORD	3
PROGRAMSKI ODBORI / PROGRAMME COMMITTEES	4
Conceptual Design of a Decision Support Tool for Severe Accident Management in Nuclear Power Plants / Bohanec Marko, Vrbanic Ivan, Bašic Ivica, Debelak Klemen, Štrubelj Luka	5
Analiza značilnosti za napovedovanje poslabšanja kroničnega srčnega popuščanja / Gjoreski Martin, Gradišek Anton, Budna Borut, PI3	9
Object Detection Overview / De Masi Carlo M., Luštrek Mitja	13
EkoSmart asistent za iskanje po integracijski platformi / Golob David, Gams Matjaž, Tavčar Aleš	17
Virtual Assistant Aggregator for the Project Electronic and Mobile Health / Kocuvan Primož, Tavčar Aleš, Gams Matjaž, Grasselli Gregor	21
Predictive Modelling of Feeling of Health for Congestive Heart Failure Patients / Valič Jakob, Mlakar Miha, Budna Borut, Luštrek Mitja	25
Categorising Behavioural States of People with Profound Intellectual and Multiple Disabilities / Cigale Matej, Slapničar Gašper, Luštrek Mitja	29
Izdela govorne zbirke za sintezo slovenskega govora / Šef Tomaž, Romih Miro, Žganec Gros Jerneja	33
Handling Real-World Problems within the COCO Platform / Tušar Tea, Brockhoff Dimo, Volz Vanessa, Hansen Nikolaus	37
Risk Stratification of Cardiac Patients by Utilizing Additional Unlabeled Examples / Papič Aleš, Bosnić Zoran	41
Fitness-Based Student Clustering Combining Clustering Algorithms and Dimensionality Reduction / Dovgan Erik, Luštrek Mitja	45
Counting Bites with a Smart Watch / Luštrek Mitja, Reščič Nina, Janko Vito, Fele Benjamin	49
Austrian-Slovenian Intelligent Tourist-Information Center Platform / Zupančič Jernej, Grasselli Gregor, Gams Matjaž	53
Razvoj postopka diarizacije govorcev z algoritmi strojnega učenja / Ktrašnik Marko, Lukan Junoš, Luštrek Mitja, Štruc Vitomir	57
Measurement of Bradykinesia for the Detection of Parkinson's Disease / Čebokli Edi, Georgiev Dejan, Žabkar Jure	61
CmoPy: Constrained Multiobjective Optimization in Python / Vodopija Aljoša, Filipič Bogdan	65
Designing an Intelligent Cognitive Assistant for Behavior Change in Mental Health / Kolenik Tine, Gjoreski Martin, Gams Matjaž	69
Classifying Power Quality Disturbances in Noisy Conditions using Machine Learning / Velichkovska Bojana, Markovska Marija, Gjoreski Hristijan, Tashkovski Dimitar	73
EMZ – Elektronsko in mobilno zdravstvo / Tavčar Aleš, Bizjak Jani, Gams Matjaž	77
Five Attempts at Cross-Dataset Speech Emotion Recognition / Luštrek Mitja, Andova Andrejaana	81
Indeks avtorjev / Author index	85

Zbornik 22. mednarodne multikonference
INFORMACIJSKA DRUŽBA – IS 2019
Zvezek A

Proceedings of the 22nd International Multiconference
INFORMATION SOCIETY – IS 2019
Volume A

Slovenska konferenca o umetni inteligenci
Slovenian Conference on Artificial Intelligence

Uredili / Edited by

Mitja Luštrek, Matjaž Gams, Rok Piltaver

<http://is.ijs.si>

10. oktober 2019 / 10 October 2019
Ljubljana, Slovenia

PREDGOVOR

Leto 2019 obeležuje petdesetletnica International Joint Conference on Artificial Intelligence, ki velja za najpomembnejši svetovni dogodek na področju umetne inteligence. Število prispevkov na konferenci se v zadnjem desetletju nenehno povečuje in je v zadnjih dveh letih zrastle za več kot 30%. Letos je bilo oddanih približno 5000 prispevkov, sprejetih je bilo 850 delež sprejetih pa je bil z manj kot 18% eden najnižjih doslej. Število referatov iz Kitajske (38%) je preseglo število referatov iz ZDA (169) in EU (152) skupaj, čeprav so glede nagrad, življenjskih dosežkov in vabljenih predavanj ZDA in EU še vedno prevladovali, saj se je število starejših raziskovalcev umetne inteligence v Aziji začelo povečevati šele pred kratkim.

Hiter napredek umetne inteligence je viden tudi pri številnih aplikacijah. Tako so na primer še pred nekaj 10 leti bili roboti nerodni in počasni mehanizmi, ki so se prevrnili čez vsako oviro, danes pa nekateri delajo salte in hodijo po težko prehodnem terenu. Pred desetletji so se avtonomna vozila zaletavala v parkirana in so pogosto obtičala, danes pa so najboljši avtonomni avtomobili do šestkrat varnejši kot povprečen avto s človeškim voznikom. Celo v šahu so programi globokih nevronske mreže začeli premagovati najbolj napredne preiskovalne algoritme, pri čemer so se v nekaj urah naučili igrati le iz osnovnih pravil igre. Čeprav nekatere to navdaja s strahom, je po mnenju nekaterih raziskovalcev ravno razvoj superinteligence edini možni izhod iz problemov netrajnostnega razvoja – uničujemo planet in družbo, sami pa ne uspemo najti rešitev, ker presegajo naše sposobnosti.

Zanimive dosežke umetne inteligence predstavljamo tudi na Slovenski konferenci o umetni inteligenci (SKUI), ki je naslednica konference Inteligentni sistemi, le-ta pa je bila sestavni del multikonference Informacijska družba že od njenega začetka leta 1997. Slovensko društvo za umetno inteligenco šteje SKUI za svojo konferenco. Letos je bilo sprejetih 20 prispevkov, kar je 3 več kot prejšnja leta. Podobno kot pretekla leta jih je največ z Instituta »Jožef Stefan«, nekaj pa jih je prispevala Fakulteta za računalništvo in informatiko, ki ima skupaj z Institutom vodilno vlogo pri raziskavah umetne inteligence v Sloveniji. 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. Dobrodošli so tudi prispevki iz tujine.

Mitja Luštrek, Rok Piltaver, Matjaž Gams

PROGRAMSKI ODBOR / PROGRAMME COMMITTEE

Mitja Luštrek

Matjaž Gams

Rok Piltaver

Marko Bohanec

Tomaž Banovec

Cene Bavec

Jaro Berce

Marko Bonač

Ivan Bratko

Dušan Caf

Bojan Cestnik

Aleš Dobnikar

Bogdan Filipič

Nikola Guid

Borka Jerman Blažič

Tomaž Kalin

Marjan Krisper

Marjan Mernik

Vladislav Rajkovič

Ivo Rozman

Niko Schlamberger

Tomaž Seljak

Miha Smolnikar

Peter Stanovnik

Damjan Strnad

Peter Tancig

Pavle Trdan

Iztok Valenčič

Vasja Vehovar

Martin Žnidaršič

Conceptual Design of a Decision Support Tool for Severe Accident Management in Nuclear Power Plants

Marko Bohanec
Jožef Stefan Institute
Dept. of Knowledge Technologies
Jamova cesta 39
SI-1000 Ljubljana, Slovenia
marko.bohanec@ijs.si

Ivan Vrbanić, Ivica Bašić
APoSS
System Reliability and Safety Analyses
Repovec 23B
HR-49210 Zabok, Croatia
ivan.vrbanic@zg.t-com.hr
basic.ivica@kr.t-com.hr

Klemen Debelak, Luka Štrubelj
GEN energija, d.o.o.
Trg republike 3
SI-1000 Ljubljana, Slovenia
{klemen.debelak,
luka.strubelj}@gen-energija.si

ABSTRACT

Safety is one of the major concerns connected with the operation of nuclear power plants. Severe accidents are very rare, but may cause very large consequences. The prevention and management of accidents requires carefully designed safety plans, guidelines and decision support tools. In this paper, we present a conceptual design of a decision support system for severe accident management. The system is aimed at providing essential information to the accident management team, in terms of the assessment of the damage state, prediction of possible progressions of the accident, and assessment of available management actions and their consequences. The system will employ components and models developed through probabilistic safety assessment and qualitative multi-criteria decision modeling. The software is being developed in the context of the EU H2020 project NARSIS. Its first prototype is expected in 2020.

Keywords

Nuclear Power Plant Safety, Severe Accident Management, Decision Support System, Probabilistic Safety Assessment.

1. INTRODUCTION

Electric energy is an indispensable resource in the modern world. Global electricity demand in 2018 increased by 4%, or 900 TWh, growing nearly twice as fast as the overall demand for energy [1]. About 11% of the world's electricity is generated by about 450 nuclear power plants (NPPs), and about 60 more are under construction [2]. NPPs provide high power output with relatively low operational costs and low impacts on the environment [3]. In the unlikely case of a severe accident, NPP may, however, cause significant and long-term consequences to people, the environment or the facility [4]. Those may include casualties, severe effects on individual's health, emitting radioactive isotopes to the environment, or melting the reactor core [5]. In spite of the remarkable reliability of current NPP safety procedures, the 2011 Fukushima Daiichi accident highlighted a number of challenging issues [6], among others the need (1) to seek out and act on new information about hazards, including combined natural events (such as the earthquake and tsunamis in Fukushima), and (2) improve nuclear plant systems, resources, and training to enable effective responses to severe accidents.

Some of these challenges are addressed in an ongoing EU H2020 project NARSIS (*New Approach to Reactor Safety ImprovementS*) [7]. The project involves 18 partners from 9 European countries and aims to improve the current Probabilistic Safety Assessment (PSA) procedures by elements that take into

account coincidental external events, vulnerability of the elements to complex aggressions, and better treatment of uncertainties through adoption of probabilistic framework for vulnerability curves and non-probabilistic approach to constraining the "expert judgments". PSA is, together with its deterministic counterpart, DSA, the main analytical method used for assessing nuclear safety, which allows practitioners to better understand the causes that can initiate nuclear accidents and to identify the most critical elements of the systems [7].

One of the goals of NARSIS is to develop a prototype Decision Support Tool for Severe Accident Management (hereafter called *Severa*). *Severa* will be aimed at supporting the NPP Technical Support Center (TSC), which is responsible for managing severe accidents through the assessment of the current situation, identifying and assessing available management actions and their consequences, selecting the actions, and monitoring the NPP response. In this stressful and complex decision situation, *Severa* will, based on measurements of operational parameters and using various insights or information from PSA models, assess the damage state of NPP barriers, predict possible progressions of the accident, and assess the available management actions. In this way, it will help the TSC to select the most appropriate management actions in a given situation, considering the likelihood of their successful implementation and possible impacts on the NPP and its environment.

Severa is currently under development, to be completed in 2020. In this paper, we present its conceptual design. We first describe the addressed decision problem. In section 3, we formulate the requirements for *Severa* and describe its intended use in a decision-making loop. In section 4, the main building blocks of *Severa* are proposed and described. Section 5 concludes the paper.

2. PROBLEM DESCRIPTION

A *severe accident* is characterized by circumstances that can cause severe core degradation, damage to the nuclear fuel, reactor pressure vessel and the NPP containment structures, possibly leading to a release of radioactivity to the environment. A severe accident may be initiated by internal (e.g., multiple equipment failures) or external (e.g. natural hazard) events, leading to NPP damage states that cannot be handled by normal operation procedures. In this case, a timely, accurate and well-justified management response is essential for preventing and mitigating the consequences of the event. For such situations, modern NPPs provide an extensive set of Severe Accident Management Guidelines (SAMGs), carefully designed and frequently reviewed written procedures for mitigating severe accidents.

SAMGs are meant to be activated in rare cases when critical parameters considerably exceed the normal operating values, for instance in the case of an unlikely accident scenario, under which the core exit temperature would exceed 650 °C. In such a case, the TSC would be formed, and the responsibility for accident management transferred from the operators in the control room to the TSC, so that the SAMGs would be used by the TSC members. The TSC would then be faced with the situation in which it has to diagnose the NPP status and recommend a sequence of management actions. Depending on the type of the accident, decisions may be time-critical and may need to be made in time windows measured in minutes after the accident.

The NPP is a complex system, and such would be decision making in real situations. At a very basic level, however, we consider that the NPP contains various barriers that prevent the emission from radioactive fuel and debris into the environment. The most important are three barriers: (1) Cladding of fuel in the Reactor Core (RC), (2) Reactor Coolant System (RCS), and (3) Containment. Accident management actions strive to prevent or minimize structural or operational damage to these barriers, for instance by identifying possible actions to decrease the core temperature or reactor pressure vessel and containment pressure. The main management strategies thus include actions such as (1) inject water into the steam generator, (2) depressurize the RCS, (3) inject water into the RCS and control containment conditions (pressure and temperature). The actual implementation of these strategies may vary depending on the current NPP status, available resources (e.g., external power supply), available equipment (e.g., mobile water pumps) and available staff. In a given situation, some of the actions may be unfeasible, may have negative effects on accident progression, or cannot resolve the problem within the required time constraints. All these factors, together with potentially severe consequences of wrong decisions, lead to an extremely complex decision problem that poses a large burden on the TSC team.

3. REQUIREMENTS FOR SEVERA

The objective of Severa is to provide an effective software tool for decision-support in the NPP severe accident management, relying on the PSA techniques and the current status of SAMGs. Severa will address the decision-analysis and decision-support needs of the TSC, once it has been formed and SAMGs have been activated. Severa is expected to support the following functions:

- Provide means to represent, store and monitor selected physical measurements of the NPP.
- Assess the current state of the vital NPP barriers: Core, RCS/Reactor Vessel, and Containment.
- Predict the future accident progression in the case that no action is undertaken by the TSC.
- Provide a list of possible management recovery strategies and courses of actions.
- Assess the applicability and feasibility of possible actions in the given situation.
- For each action: predict the consequences in terms of probability of the last barrier (containment) failure and estimated time window for failure.
- Evaluate and rank the feasible actions, providing recommendations for the TSC.

In the framework of NARSIS, only a prototype implementation of Severa is foreseen. It will be used mainly to demonstrate the feasibility of developing such a tool for actual NPPs and its

potential for managing and reducing the residual risk from NPPs operation. It may also improve the training of TSC members. Due to the complexity of the decision problem, Severa will be further restricted to reasonably small, but relevant, subsets of system parameters and management actions. Severa's performance will be tuned and measured on the "Virtual NPP": an artificial, somewhat simplified, but sufficiently realistic NPP architecture, defined in the NARSIS project for research and testing purposes.

Severa is foreseen to be used in repeated decision-making cycles. Each cycle is expected to take about 10 to 20 minutes and will consist of the following main steps:

1. Monitor and assess the NPP status: relevant parameters (e.g., primary system or containment), and availability and performance of plant systems.
2. Collect the information concerning the current status of plant damage and accident progression; e.g., which barriers are challenged or may be soon, which functions are not available.
3. Identify possible alternatives (action courses); identified action courses should include the actions which are required by the SAMGs and should consider the availability of plant systems/functions and time window necessary for the action.
4. For each identified alternative establish the answers to the set of plant status questions, i.e., establish the input for the tool for each alternative action.
5. Quantify (assess, evaluate) each alternative using the models implemented in the tool.
6. Compare the alternatives based on the results from the tool and select the alternative to proceed with.
7. Implement the selected actions and observe plant's response.

4. CONCEPTUAL DESIGN

At the highest level, Severa will consist of two parts (Figure 1):

- I. *Monitoring*: Observing and assessing the situation "as-is", without any human intervention.
- II. *Management*: Facilitating the decision-analysis and decision-support activities of the TSC.

4.1 Monitoring

The objective of the monitoring part is to provide information about the current state of important NPP barriers and possible progressions of events. It consists of three modules: Input Parameters (IP), Diagnostic Module, and Prognostic Module.

4.1.1 Input Parameters

This module provides means to store and manage a time series of physical parameters, which are measured at critical operational points in the NPP:

- CET: Core Exit Thermocouples [°C]
- SGL: Steam Generator Level [m]
- RPVL: Reactor Pressure Vessel Level [%]
- Prcs: Reactor Coolant System Pressure [MPa]
- Pcont: Containment Pressure [MPa]
- Tcont: Containment Temperature [°C]
- H2: Hydrogen concentration [%]

Each parameter is represented in terms of a numerical measurement (e.g., SGL = 8.9 m, H2 = 0.21 %) and color-coded severity level (green, yellow, orange, red). The latter is determined by a discretization of the former. The orange and red levels denote a severe situation and generally require the activation of SAMGs.

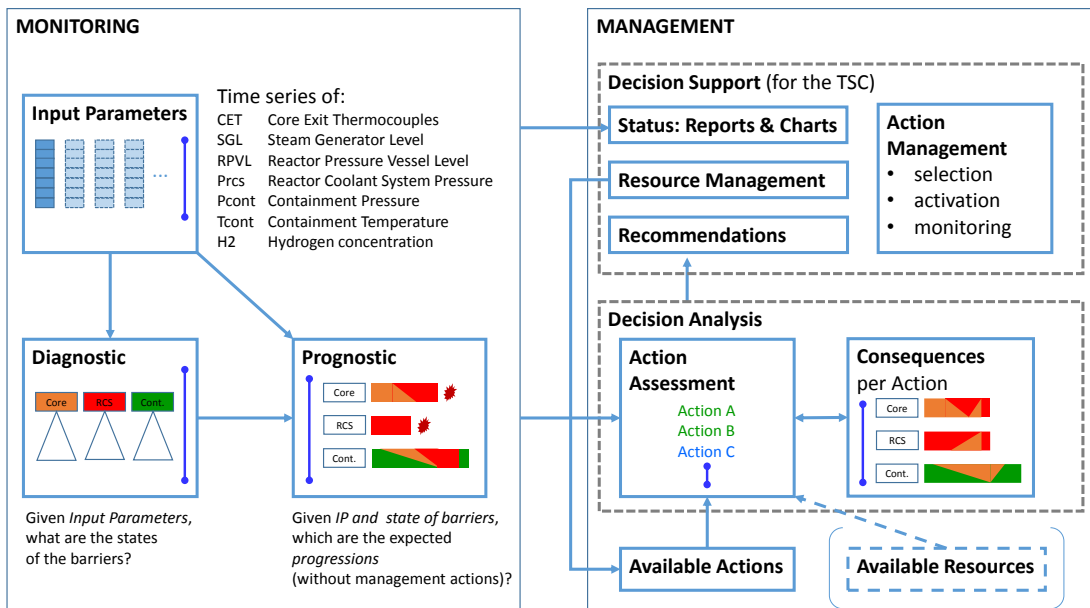


Figure 1: Building blocks of Severa. The symbol  indicates the current time.

Even though these parameters can be measured continuously, a typical time granulation is expected to be about 10 to 20 minutes, corresponding to the expected time cycle of using the system.

4.1.2 Diagnostic Module

The purpose of this module is to determine the status of each of the three barriers: Core, RCS (vessel) and Containment. Each barrier has an individual set of possible states. For instance, states of the Core are OX (cladding oxidation), CD (core damage), EX (corium ex-vessel) or OK. For comprehensibility, the states are color-coded, too.

Barrier states are determined from the values of IP. They form a time series, which corresponds to the granularity of the IP, i.e., each set of barrier states corresponds to each IP vector.

Employed methods: Basic information about determining barrier states will be provided in terms of tables and/or decision trees. In Severa, states will be determined using DEX hierarchical models [9]: one model for each barrier. Decision rules in the DEX models will facilitate probabilistic assessment of states [9], for instance in cases of missing or inaccurate IP measurements.

4.1.3 Prognostic Module

Given the IP and outcomes of the Diagnostic Module, the purpose of the prognostic module is to predict the future progression of the barrier states if no management actions are undertaken. The progression information include:

- Identification of the future state(s) for each barrier.
- Estimation of time until the next state change.
- Time series of probabilistic distributions of predicted future states.

Methods: A combination of PSA modelling methods, primarily event trees and probabilistic assessment models [10].

4.2 Management

The management part of Severa will provide decision-analysis and decision-support functionality for the TSC. The main purpose is to inform the TSC about the current state of the NPP and to give advice about possible recovery actions and their consequences.

The management part of the system is particularly concerned with *management actions*. These represent possible decision alternatives, which are at the TSC's disposal at a given situation and time frame. The TSC has to consider the possible actions and assess their feasibility and consequences with respect to the integrity of the barriers. The TSC may select an action (possibly leading to a series of actions), allocate the necessary resources, and monitor the progress while the action is being carried out.

To help performing these activities, Severa will provide databases of available actions and resources, and two modules, called Decision Analysis Module and Decision Support Module.

4.2.1 Database of Available Actions

This database provides a collection of possible recovery actions envisioned in the SAMGs. Only a subset of actions may be feasible in a given situation. Thus, each action has associated a number of properties (or even models), aimed at determining:

- *Applicability* (relevance, entry conditions): Is the action relevant for recovering the current situation?
- *Feasibility*: Is the action feasible given the current IP, barrier states, available resources and available time for recovery?
- Possible *impacts* and *consequences* in terms of probability of barrier (containment) failure in a given time window.

4.2.2 Database of Available Resources

This is a foreseen collection of resources available for carrying out any actions, including material resources (electric energy supply, availability of pumps and generators, etc.) and human resources. The availability of resources primarily influences the feasibility, completion time and expected success of actions. When an action is activated, the corresponding resources are allocated or spent.

Because resources are usually vast and very specific to individual NPPs, this database will not be implemented explicitly in the Severa prototype; the availability of resources will be modelled implicitly through manual input to the decision analysis module.

4.2.3 Decision Analysis Module

Among all the available management actions, only a few of them may actually be relevant in a given situation. The purpose of the Decision Analysis Module is to:

- Identify actions that are applicable and feasible in the current situation.
- Predict possible consequences in terms of probability of barrier failure and assessed time window for failure.
- Assess the quality of actions according to multiple criteria.
- Make a priority ranking of actions to be recommended to the TSC.

In principle, the consequence prediction will be done similarly as in the Prognostic Module, that is, in a reusable way. The difference is that the Prognostic Module assumes no actions and the Decision Analysis Module depends on a specific action. The unification is foreseen by defining a special action *status-quo*, to be used in the diagnostic stage.

Methods: Actions will be described in terms of multiple parameters, whose probabilistic values will be assessed through a series of plant status and phenomenological questions [10]. An APET (Accident Progression Event Tree) [10], implemented as a DEX probabilistic model [9], will be employed to assess the expected probabilities of eventual radioactive emissions.

4.2.4 Decision Support Module

This module directly supports the decision-making process of the TSC, providing the following functionality:

- *Presenting* all the relevant information that comes from the other parts of the system, in a transparent and user-friendly way, mainly using charts and reports. This information includes:
 - *IP values* presented in tables and charts,
 - *diagnostic* and *prognostic* information from the monitoring subsystem,
 - *action recommendations* in terms of action ranking and assessment of consequences.
- *Resource management:* Allows the TSC to provide additional input data about resources that is needed in order to assess the feasibility and expected consequences of actions. For instance, the TSC may indicate that some resources are unavailable and cannot be used in recovery actions. For reasons that cannot be measured or determined by the system, the TSC may also manually enable or disable specific actions.
- *Action Management:* Actual selection, activation and monitoring of actions.

5. CONCLUSION

The immediate goal of the decision support system, whose conceptual design has been presented above, is to demonstrate the feasibility of developing such a tool, using the PSA and decision-modeling methods, for actual NPPs. The approach is novel in attempting to operationalize SAMGs for a specific severe accident situation, focusing on decision-support needs of the TSC.

Ultimately, such tools are expected to substantially contribute to severe accident management in NPPs as tools for the training of TSC members and as decision-support tools in real situations. The main contributions include:

- providing a timely and best available information about the state of NPPs barriers and potential future developments of the event;

- information support to the TSC team in an extremely difficult and stressful decision situation;
- operationalization of SAMGs for a specific situation;
- reducing the risk of overlooking important management actions and, consequently, making wrong decisions.

Further work will be focused on the implementation of the system (to be completed in 2020) and its experimental evaluation using the NARSIS' specification of the "Virtual NPP". The evaluation will address both the aspects of verification (compliance of Severa with the SAMGs) and validation (meeting the needs of the TSC in severe accident management). The latter will be assessed by comparing the performance of the TSC on selected accident scenarios with and without the support of Severa.

6. ACKNOWLEDGMENTS

This work has been carried out within the NARSIS project [7], which has received funding from the European Union's H2020-Euratom Program under grant agreement N° 755439.

7. REFERENCES

- [1] Global Energy & CO2 Status Report, <https://www.iea.org/geco/electricity/>
- [2] Nuclear Power in the World Today, <https://www.world-nuclear.org/information-library/current-and-future-generation/nuclear-power-in-the-world-today.aspx>, Updated February 2019.
- [3] Nuclear Energy: Pros & Cons, <https://www.renewableresourcescoalition.org/nuclear-energy-pros-cons/>
- [4] Nuclear and radiation accidents and incidents, https://en.wikipedia.org/wiki/Nuclear_and_radiation_accidents_and_incidents
- [5] Staff, IAEA, AEN/NEA. International Nuclear and Radiological Events Scale Users' Manual, 2008 Edition (PDF). Vienna, Austria: International Atomic Energy Agency. p. 184. Archived from the original (PDF) on May 15, 2011. Retrieved 2010-07-26.
- [6] Committee on Lessons Learned from the Fukushima Nuclear Accident for Improving Safety and Security of U.S. Nuclear Plants; Nuclear and Radiation Studies Board; Division on Earth and Life Studies; National Research Council. Washington (DC): National Academies Press (US); 2014.
- [7] NARSIS, *New Approach to Reactor Safety ImprovementS*, EU Horizon 2020 project H2020-RIA-755439 (2017-2020) <http://www.narsis.eu/>
- [8] Bohanec, M.: *DEXi: Program for Multi-Attribute Decision Making, User's Manual*, Version 5.00. IJS Report DP-11897, Jožef Stefan Institute, Ljubljana, 2015. <http://kt.ijs.si/MarkoBohanec/pub/DEXiManual500p.pdf>
- [9] Trdin, N., Bohanec, M.: Extending the multi-criteria decision making method DEX with numeric attributes, value distributions and relational models. *Central European Journal of Operations Research* 26(1), 1–41, DOI: 10.1007/s10100-017-0468-92018, 2018.
- [10] Vrbanić, I., Bašić, I., Štrubelj, L., Debelak, K., Bohanec, M.: *Definition of Hazard-Induced Damage States and Development of State-Specific APETs for Demonstration Purposes*, NARSIS Deliverable D5.3, 2019

Analiza značilik za napovedovanje poslabšanja kroničnega srčnega popuščanja

Feature analysis for detecting different chronic heart failure stages

Martin Gjoreski, Anton Gradišek, Borut Budna
Odsek za inteligentne sisteme, Institut Jožef Stefan
martin.gjoreski@ijs.si

Gregor Poglajen
Univerzitetni klinični center Ljubljana
gregor.poglajen@ukclj.si

POVZETEK

Kronično srčno popuščanje, pri katerem srce ne more črpati dovolj krvi, da bi zadostilo potrebam telesa, spada med kronične bolezni, ki pogosto prizadenejo predvsem starejše od 65 let. Ob poslabšanju stanja je pogosto potrebna hospitalizacija, kar predstavlja breme tako za pacienta kot za zdravstveni sistem. Naš cilj je razviti telemedicinsko metodo, ki na podlagi analize srčnih tonov z metodami strojnega učenja prepozna poslabšanje. Tu predstavimo začetno analizo značilik na podlagi zvoka pri 22 pacientih. Primerjali smo posamezne paciente v dveh fazah – v dekompenzirani in rekompenzirani fazi. Identificirali smo značilke, pri katerih je vrednost v eni od faz skoraj vedno večja od vrednosti v drugi fazi. Ta trend se je pokazal pri 20 od 22 pacientov (91 %). Te značilke nam bodo lahko v bodoče pomagale pri gradnji personaliziranih modelov za zaznavanje različnih faz kroničnega srčnega popuščanja.

Ključne besede

Srčno popuščanje, analiza zvoka, strojno učenje.

ABSTRACT

Chronic heart failure, where the heart cannot pump enough blood to supply the body's needs, is a chronic disease that affects especially people above 65. Worsening of the condition often requires hospital treatment, which is a burden both for the patient and for the health system. Our aim is to develop a telemonitoring method that will use advanced machine-learning methods on sound recordings to detect onset of worsening. Here, we present the initial analysis of sound-based features using recordings of 22 patients. The analysis is performed by comparing individuals in two stages of heart failure – in decompensated and recompensated phase. We identified features for which the value in the decompensation phase is almost always bigger compared to the value of the same feature in the recompensated phase. This trend was present for 20 out of 22 pairs patients (91% of the patients). Thus, these features may be useful for building personalized machine learning models for detecting decompensated and recompensated stages in CHF patients.

Keywords

Chronic heart failure, sound analysis, machine learning.

1. UVOD

Kronično srčno popuščanje (SP, angleško Chronic Heart Failure, CHF) je ena od epidemij sodobnega časa. Gre za kronično progresivno bolezen, kjer srce ni sposobno črpati dovolj krvi, da bi zadostilo potrebam telesa, kar se pri pacientih izraža kot izrazitejšo poslabšanje telesne zmogljivosti, težje dihanje ter otekanje v okončine in/ali trebuh. Bolezen v razvitem svetu prizadene 1-2 % splošne populacije ter do 10 % populacije, starejše od 65 let. Po ocenah trenutno (podatek iz leta 2015) živi na svetu več kot 23 milijonov bolnikov s SP, od tega več kot 5 milijonov v ZDA. Študije ocenjujejo, da tam strošek za skrb za bolnike znaša 31 milijard dolarjev, kar bo do leta 2030 naraslo na 70 milijard [1][2].

V kliničnem poteku SP se tipično izmenjujejo obdobja dobrega počutja (tj. kompenzirane faze) in obdobja poslabšanja SP (tj. faze dekompenzacije), ko pacient opaža zgoraj opisane simptome. Ob poslabšanju SP je pogosto potrebna hospitalizacija, kjer se bolnikom pomaga z intravenoznimi zdravili. Če se poslabšanje SP zazna dovolj zgodaj, lahko z ustreznimi terapevtskimi ukrepi, kot so omejitve vnosa tekočine in povečanjem odmerka diuretikov, bolniku prihranimo hospitalizacijo.

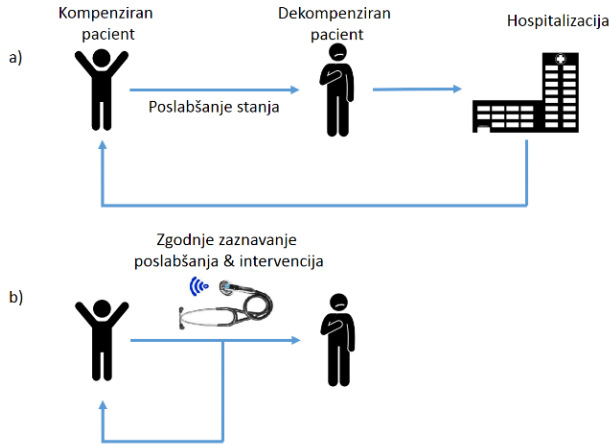
Izkušen zdravnik lahko poslabšanje SP prepozna ob pregledu bolnika, pogosto se poslabšanje izraža kot sprememba oz. pojavljanje dodatnih srčnih tonov (tipično tretji ton, ki se pojavi 0,1-0,2 s za drugim tonom), kar zdravnik spremlja s stetoskopom (fonokardiografija). Raziskave v zadnjem času kažejo, da se nekateri fiziološki parametri (npr. dodatni srčni toni, porast tlakov v pljučnem krvotoku) pričnejo spreminjati že nekaj tednov preden pride do klinično očitnega poslabšanja SP. Prve analize [3] že kažejo, da zgodnje ugotavljanje spreminjanja teh parametrov lahko pomeni precej učinkovitejšo obravnavo SP, saj je epizod polno razvitega poslabšanja srčnega popuščanja manj, s tem pa tudi manj potrebe po hospitalni obravnavi.

V pričujočem prispevku raziskujemo možnosti zaznavanja poslabšanja SP na podlagi analize srčnih tonov z metodami umetne inteligence. Raziskava je del daljšega projekta, cilj katerega je razviti metodo ali telemedicinsko napravo, s katero bo uporabnik lahko sam spremljal svoje zdravstveno stanje in ob nakazanem poslabšanju SP kontaktiral zdravnika, kar shematsko prikazuje Slika 1.

V preteklih analizah [4][5] smo razvili metodo, s katero uspešno ločujemo med posnetki srčnih tonov zdravih posameznikov ter bolnikov v dekompenzirani fazi SP. Metoda je sestavljena iz filtriranja, segmentacije, izračuna značilik in gradnje klasifikacijskih modelov strojnega učenja. Klasifikatorje smo kombinirali z uporabo pristopa zlaganja (stacking), kjer so osnovni

klasifikatorji klasificirali posamezne segmente, meta-klasifikator pa je klasificiral celotne posnetke, saj vemo, da vsi segmenti nekega posnetka pripadajo istemu razredu. Metoda je v primerjavi z večinskim klasifikatorjem dosegla 15 % višjo natančnost.

V pričujočem prispevku se ukvarjamo z vprašanjem personalizacije. Zanima nas, če lahko ločujemo med posnetki pri posameznikih v različnih fazah SP. Na tej stopnji primerjamo posnetke pacientov v dekompenzirani fazi, ob sprejetju v bolnišnico, ter v rekompensirani fazi, ko je pacient odpuščen iz bolnišnice.



Slika 1: Shematski prikaz razvoja poslabšanja srčnega popuščanja. V zgornjem primeru mora pacient zaradi poslabšanja SP v bolnišnico (hospitalizacija), v spodnjem primeru poslabšanje zaznamo dovolj zgodaj, da lahko z ustrezno intervencijo preprečimo poslabšanje in se hospitalizaciji izognemo.

2. MATERIALI IN METODE

2.1 Baza podatkov

V študiji smo obravnavali posnetke 22 bolnikov s kroničnim srčnim popuščanjem, ki so bili hospitalizirani zaradi poslabšanja zdravstvenega stanja. Prvi posnetek je nastal v dekompenzirani fazi (ob hospitalizaciji), drugi pa v rekompensirani fazi, ob odpustu iz bolnišnice.

Za snemanje smo uporabili profesionalni medicinski digitalni stetoskop 3M Littmann Electronic Stethoscope Model 3200. Posnetek, fonokardiogram, je bil vedno sneman na Erbovi točki, nad tretjim medrebrnim prostorom levo od grodnice. Posamezen posnetek je dolg do 30 s, kar je omejitvev stetoskopa.

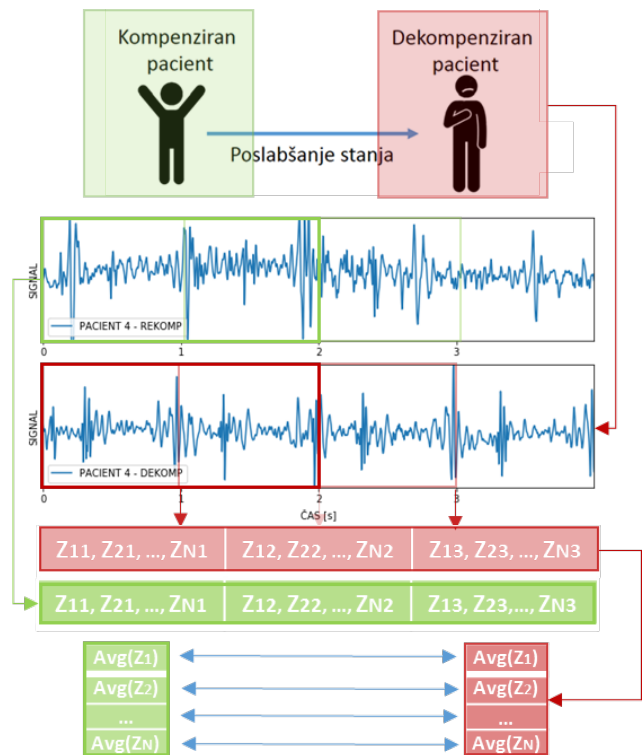
Pred začetkom študije smo pridobili pozitivno mnenje Komisije za medicinsko etiko.

2.2 Metoda

Za analizo smo uporabili orodje za klasifikacijo zvokov [6], ki smo ga razvili na platformi za delo s podatki Orange. Orodje za ekstrakcijo značilk uporablja programsko opremo OpenSMILE [7], ki je bila prvotno razvita leta 2009 za prepoznavanje čustev na podlagi zvoka, vendar se je kasneje začela uporabljati tudi v bolj splošne namene, kot je pridobivanje informacij iz glasbenih posnetkov.

Shema metode je predstavljena na Sliki 2. Za vsakega pacienta smo imeli 2 posnetka, enega v rekompensirani fazi (zeleno) in drugega v dekompenzirani fazi (rdeča). Najprej smo vse posnetke segmentirali z uporabo tehnike drsečega okna. Na Sliki 2 imamo posnetek dolžine dveh sekund. Za velikost drsečega okna dve sekundi in 50 % prekrivanja (dve sekundi) smo dobili za vsak posnetek tri segmente. Iz vsakega segmenta smo izračunali 1941 zvočnih značilk, med njimi statistične značilke (npr. varianco, koeficient simetrije, sploščenost), energijske značilke (npr. energije v intervalu med 250 Hz in 1 kHz), frekvenčne značilke (spektralna odstopna mesta pri 25 %, 50 %, 75 % in 90 %) in glasovne značilke (npr. tresenje, barva..., razmerje signal/šum). Celoten seznam značilk je opisan v članku Schuller in ostali [8]. Na Sliki 2 so značilke označene kot: $Z_{11}, Z_{21}, \dots, Z_{N1}$, kjer prvi indeks označuje značilko (N značilk), drugi pa segment, na katerem je bila specifična značilka izračunana. V nadaljevanju se izračuna povprečne vrednosti značilk čez vse segmente (na Sliki 2 imamo povprečje treh segmentov). Tako dobimo za vsak posnetek vektor značilk: $\text{Avg}(Z_1), \text{Avg}(Z_2), \dots, \text{Avg}(Z_N)$.

Za izbiro značilk, ki so najbolj primerne za ločevanje med obema fazama SP, smo uporabili Wilcoxonov statistični test [9], ki spada med neparametrične teste in testira, kdaj dva povezana vzorca izvirata iz iste distribucije. Statistične teste smo izvedli za okna velikosti: 10, 8, 6, 4, 2 in 1 sekunde s 50 % prekrivanjem. V Tabeli 1 je predstavljeno število značilk, za katere je bila vrednost p manjša kot 0.001. Iz rezultatov lahko razberemo, da imamo za vsako velikost okna najmanj sedem informativnih značilk. Poleg tega vidimo, da število informativnih značilk narašča ob krajšanju dolžine okna. Poskusi kažejo, da je optimalna velikost okna za našo podatkovno zbirko dve sekundi s 50 % prekrivanjem.



Slika 2: Shema metode.

Tabela 1: Število informativnih značilk, ki ločijo med rekompenzirano in dekompenzirano fazo, v odvisnosti od dolžine okna.

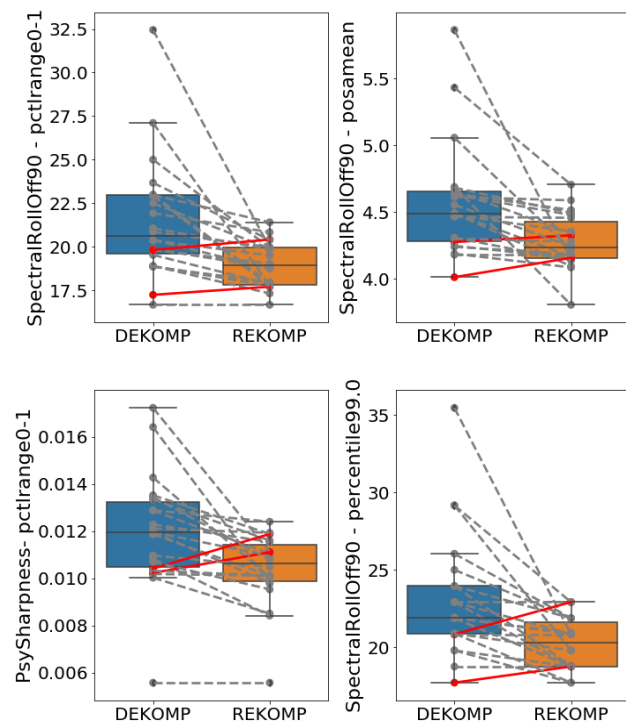
Velikost okna (v sekundah)	10	8	6	4	2	1
# informativnih značilk	7	11	14	14	16	14

Slika 3 predstavlja škatlaste diagrame za štiri naključno izbrane značilke izmed 16, ki smo jih dobili z uporabo okna dolžine 2 s.

Štiri značilke, ki jih prikazemo v tem primeru, so sledeče:

- `fftMag_spectralRollOff90.0_sma_de_percentile1.0`
- `fftMag_spectralRollOff90.0_sma_de_percentile99.0`
- `fftMag_spectralRollOff90.0_sma_de_posamean`
- `fftMag_psySharpness_sma_de_pctlrange0-1`

Vse značilke so povezane s spektralnimi lastnostmi segmenta. Prve tri značilke izračunajo Fourierovo transformacijo na kratkih oknih, izvedejo glajenje z drsečim oknom, izračunajo prvi odvod signala po času in določijo statistične označevalce, *posamean* pa predstavlja pozitivno aritmetično sredino. Značilka *psySharpness* se nanaša na spektralno strukturo signala, pri signalih, ki vsebujejo visoke frekvence, je vrednost te značilke višja. Podrobneje so te značilke opisane v [8].



Slika 3: Škatlasti diagrame za štiri značilke, ki jih statistični testi izberejo kot najbolj pomembne. Modra škatla predstavlja dekompenzirano fazo, oranžna rekompenzirano. Sive črte povezujejo vrednosti značilk za posamezne paciente, kjer je vrednost v dekompenzirani fazi večja od tiste v rekompenzirani, rdeče pa tiste, pri katerih velja obratno.

Modra škatla na Sliki 3 prikazuje vrednosti značilk izračunanih iz dekompenziranih posnetkov, oranžna pa vrednosti značilk iz rekompenziranih. Siva črta ponazarja mejo, kjer je vrednost rekompenzirane značilke manjša kot pripadajoča vrednost dekompenzirane značilke, rdeča črta pa ravno obratno. Iz slike je razvidno, da imamo za posamezno značilko zgolj dve rdeči črti. Če torej izvedemo binarno klasifikacijo na podlagi samo ene poljubne značilke izmed teh štirih, bomo napačno razvrstili dva izmed 22 parov posnetkov.

Prostora za izboljšave je sicer še veliko. Načeloma bi za klasifikacijo lahko kombinirali več značilk, vendar se izkaže, da gre pri rdečih črtah večinoma za iste posameznike – to je posledica dejstva, da so značilke korelirane med sabo. Poleg tega delamo z relativno majhnim vzorcem – na 22 posameznikih, kar bi lahko pripeljalo do tega, da bi zgradili model, ki bi bil preveč prilagojen danim podatkom.

3. ZAKLJUČEK

V prispevku prikazemo prve korake h gradnji personaliziranega modela, ki bo bolnikom omogočal spremljanje poslabšanja srčnega popuščanja. Identificirali smo vrsto značilk, ki nam lahko pomagajo pri gradnji personaliziranih modelov. Personalizacija je lahko nenadzorovana, od vsake značilke enostavno odštejem povprečje te značilke za posamezno osebo. Začetna analiza kaže, da imamo vsaj štiri značilke, za katere je vrednost v dekompenzirani fazi skoraj vedno večja od tiste v rekompenzirani (pri 20 od 22 pacientov, torej v 91 % primerov). Rezultati so obetavni in nam bodo služili pri razvoju personaliziranih modelov za zaznavanje različnih faz kroničnega srčnega popuščanja.

Čeprav se zavedamo možnosti, da je analiza preveč dobro prilagojena podatkom zaradi majhnega števila pacientov, verjamemo, da so rezultati obetavni in uporabni v prihodnje. V prihodnje nameravamo izvesti analizo na posnetkih več posameznikov, poleg tega pa bomo vključili tudi posnetke v različnih fazah, ne le v skrajnih primerih, kot v predstavljeni študiji. Če spremljamo posameznika, lahko uporabimo personaliziran model – če v določeni točki posamezne značilke, ki jih izračunamo iz zvočnih posnetkov, začnejo odstopati od pričakovanih vrednosti za rekompenzirano fazo, lahko posumimo na možnost poslabšanja.

4. VIRI

- [1] Shannon M. Duplay and Véronique L. Roger. 2014. Understanding the epidemic of heart failure: past, present, and future. *Current heart failure reports* 11, 4 (2014), 404-415.
- [2] Paul A. Heidenreich, Nancy M. Albert, Larry A. Allen, David A. Bluemke, Javed Butler, Gregg C. Fonarow, John S. Ikonomidis, Olga Khavjou, Marvin A. Konstam, Thomas M. Madox, et al. 2013. Forecasting the impact of heart failure in the United States: a policy statement from the American Heart Association. *Circulation: Heart Failure* 6, 3 (2013), 606-619.
- [3] William T. Abraham, Lynne W. Stevenson, Robert C. Bourge, Jo Ann Lindenfeld, Jordan G. Bauman, and Philip B. Adamson. 2016. Sustained efficacy of pulmonary artery pressure to guide adjustment of chronic heart failure therapy: complete follow-up results from the CHAMPION randomised trial. *The Lancet* 387, 10017 (2016), 453-461.
- [4] Martin Gjoreski, Monika Simjanoska, Anton Gradišek, Ana Peterlin, Matjaž Gams, and Gregor Poglajen. 2017. Chronic heart failure detection from heart sounds using a stack of

- machine-learning classifiers. In *2017 International Conference of Intelligent Environments (IE)*. IEEE, 14-19.
- [5] Martin Gjoreski, Anton Gradišek, Borut Budna, Matjaž Gams, and Gregor Poglajen. 2019. Toward early detection and monitoring of chronic heart failure using heart sounds. *Ambient Intelligence and Smart Environments*, 26, 336-343. DOI:10.3233/AISE190061
- [6] Martin Gjoreski, Borut Budna, Anton Gradišek, and Matjaž Gams. 2017. JSI Sound – a machine-learning tool in Orange for simple biosound classification. In *Proceedings of the Workshop on Advances in Bioinformatics and Artificial Intelligence: Bridging the Gap, co-located with 26th International Joint Conference on Artificial Intelligence (IJCAI 2017)*. Melbourne, Australia, 26-29.
- [7] Florian Eyben, Felix Weninger, Florian Gross, and Björn Schuller. 2013. Recent developments in OpenSMILE, the munich open-source multimedia feature extractor. In *Proceedings of the 21st ACM international conference on Multimedia*. ACM, 835-838.
- [8] Björn Schuller, Michel Valstar, Florian Eyben, Gary McKeown, Roddy Cowie, and Maja Pantic. 2011. Avec 2011-the first international audio/visual emotion challenge. In *International Conference of Affective Computing and Intelligent Interaction*. Springer, 415-424.
- [9] Frank Wilcoxon. 1992. Individual comparison by ranking methods. In *Breakthroughs in statistics*. Springer, 196-202.

Object Detection Overview

Carlo M. De Masi
Jožef Stefan Institute
Jamova cesta 39, 1000 Ljubljana, Slovenia
carlo.maria.demasi@ijs.si

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

ABSTRACT

In this paper, we present a brief review of some of the main algorithms adopted in the field of computer vision for the aim of object detection. We highlight the working principles of two main families of models, Region-based Convolutional Neural Network detectors (“R-CNN”) and Single-Shot object detectors (SSD, YOLO), and present a comparison between them, with a focus on the trade-off between speed and accuracy and its dependence on various models parameters.

Keywords

computer vision, object detection, CNNs

1. INTRODUCTION

Most computer vision tasks can be roughly divided into three main categories [10]:

- **image classification:** determining whether a certain class of objects is present in the image or not (Fig. 1a);
- **object detection:** determining that an object belonging to a certain class is present, and localizing it within the image (Fig. 1b);
- **semantic scene labeling:** classification of each pixel of the image as belonging to a certain class. Individual instances of the same object are usually not segmented (Fig. 1c), but modern datasets [10, 9] provide labels to distinguish between them (Fig. 1d).

In less than a decade, techniques to solve these tasks have undergone a period of extremely rapid development, most notably fueled by the successful employment of Convolutional Neural Networks (CNN) in the field.

The adoption of CNN-based algorithms was kick-started by the paper by Alex Krizhevsky, Ilya Sutskever and Geoffrey Hinton [8]. This paper showed the effectiveness of CNNs in computer vision problems, and introduced a number of techniques that are still used today, such as the adoption of ReLU as activation functions and the use of data augmentation techniques. The standard structure of CNN-based models, basically consisting of simple stacking of convolutional and pooling layers followed by one or more fully-connected layers, has been successfully used for some time in order to solve localization, object detection and human pose estimation problems [17, 20, 1, 4, 21, 19]. Improvements were generally achieved by increasing the depth and width of the

models, and by using larger amount of training data, at the cost of increasing the needed computational power and the risk of over-fitting [19]. This basic paradigm has been challenged by Google with the introduction of the “Inception” architecture [19], where parts of the network are not located sequentially, so that different operations such as pooling or convolution can be performed in parallel.

Among the many proposed architectures (see also [5, 24]), arguably one of the most impactful contributions to object detection has been given by the introduction of the Region-based Convolutional Neural Network detectors (“R-CNN”) [4] and its modification (Fast R-CNN and Faster R-CNN [3, 15]). The accuracy of these detectors, however, comes at a computational cost, which can be reduced by the adoption of the so-called Single Shot detectors.

In this paper, we present an overview of the region-based and single-shot families of models. In Section 2, we describe a few methods to identify Regions of Interest (RoIs), i.e. areas of the picture where objects are located, which is a fundamental ingredient for region-based detection algorithms. Sections 3 and 4 are dedicated to region-based and single-shot detectors, respectively. In section 5, we summarize a comparison of the performances of these algorithms, and how their trade-off between speed and accuracy is influenced by different factors.

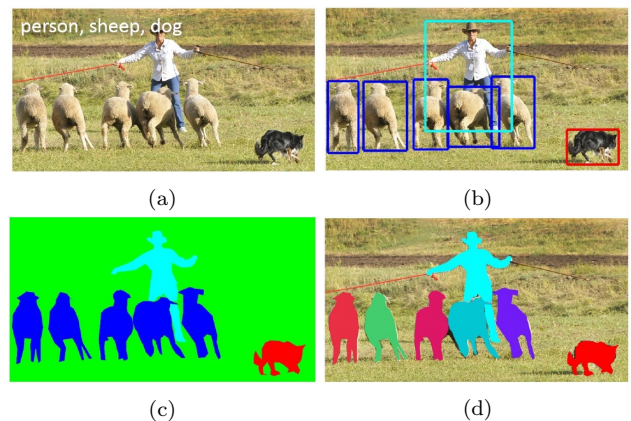


Figure 1: From [10], different types of computer vision tasks. Image classification (a), object detection (b), and semantic scene labeling (c,d).

2. REGIONS OF INTEREST (ROI) SELECTION ALGORITHMS

Region-based methods generally tackle object detection tasks by dividing it into two steps. First, it is necessary to locate the so-called Regions of Interest (RoI), i.e. areas of the image where possible objects are located. After this, the classification step takes place, where the previously located areas are classified. RoIs whose content is classified as belonging to one of the considered classes within a certain confidence are detected.

In the following, we make a short description of two of the main methods adopted to identify RoIs.

2.1 Selective search

Region proposal algorithms identify RoIs by adopting image segmentation techniques, which consist of grouping pixels based on their similarity according to some criteria. A commonly adopted method is Selective Search (SS); as shown in Fig. 2, groups of similar regions are created hierarchically, starting from single pixels, based on color, texture, size and shape compatibility [22]. A general important requirement for region proposal methods is that they should have a very high recall; false positives can then be rejected in the following classification phase.



Figure 2: From [22]. All possible ROIs during the merging.

2.2 Region Proposal Network (RPN)

Region proposal networks (RPN) are the distinctive feature of Faster R-CNNs detectors (see Section 3), and they eliminate the need to use an external algorithm for selecting RoIs. In this method, the feature map produced by the first convolutional layer of the detector is passed to a CNN that produces region proposals by predicting their bounding boxes and “objectness” scores, measuring whether each box contains an object or not [15].

3. REGION-BASED OBJECT DETECTORS (R-CNN, FAST R-CNN, FASTER R-CNN)

As mentioned earlier, region-based object detectors work in two steps, consisting of the determination of RoIs and their following classification.

The first suggested implementation of this algorithm, **R-CNN** [4], employs a region proposal method to create ≈ 2000 ROIs. As shown in Fig. 3 (top left panel), each of the identified RoIs is then resized and fed as input to a CNN,

followed by fully connected layers to classify the object and refine the boundary box.

Even though the R-CNN algorithm is very accurate, it has also the downside of being quite slow. The high number of proposals makes the algorithm slow, since each RoI is processed by the CNN separately, which means that the whole feature extraction process is repeated 2000 times.

This limitation has been overcome in **Fast R-CNNs** [3], as depicted in Fig 3 (bottom left panel). In this architecture, features are extracted only once for the whole image by using a CNN, while an external region proposal method (such as selective search) is used to create RoIs. After this step, the feature map and the RoIs are combined, producing patches that, as in the previous case, are resized (RoI pooling layer) and passed as input to a fully connected layer for the object detection.

In the case of Fast R-CNNs, the main bottleneck is caused by the use of the external region proposal method, which usually runs on a CPU and is slower than the rest of the process; out of the 2.3 seconds needed by Fast R-CNN to make a prediction in testing, ≈ 2 seconds are used for generating the 2000 ROIs [3]. So, a further improvement in the algorithm speed has been provided with the introduction of **Faster R-CNN** [15], by substituting the external region proposal method with the convolutional Region Proposal Network (RPN) we presented in Sec. 2.2 (see Fig. 3, right panel).

4. SINGLE-SHOT OBJECT DETECTORS

Differently from region proposal detectors, which perform region proposals and region classifications in two steps, single-shot detectors simultaneously predict bounding boxes and the class as they process the image in one shot.

4.1 SSD

SSD [11] is one of the fastest object detectors available. Its working principles can be summarized as follows:

- the image is passed through a series of convolutional layers, thus producing several sets of feature maps at different scales (4x4, 8x8, etc.);
- a pre-defined, default set of bounding boxes (similar to the “anchors” in RPN [15]) of different aspect ratios is provided for each location in all the produced feature maps;
- for each of these default boxes, both the offsets to the ground truth boxes and the confidence for all classes are predicted;
- default boxes are matched to ground truth boxes based on IoU (Intersection Over Union, [23]). The best predicted box is labeled a positive, along with all other boxes that have an IoU with the truth > 0.5 (see Fig. 4).

The downside of skipping region proposal is that SSD draws and classifies bounding boxes of many shapes and scales in every single position in the image, so that most of them are negative examples. For this reason, highly-overlapping

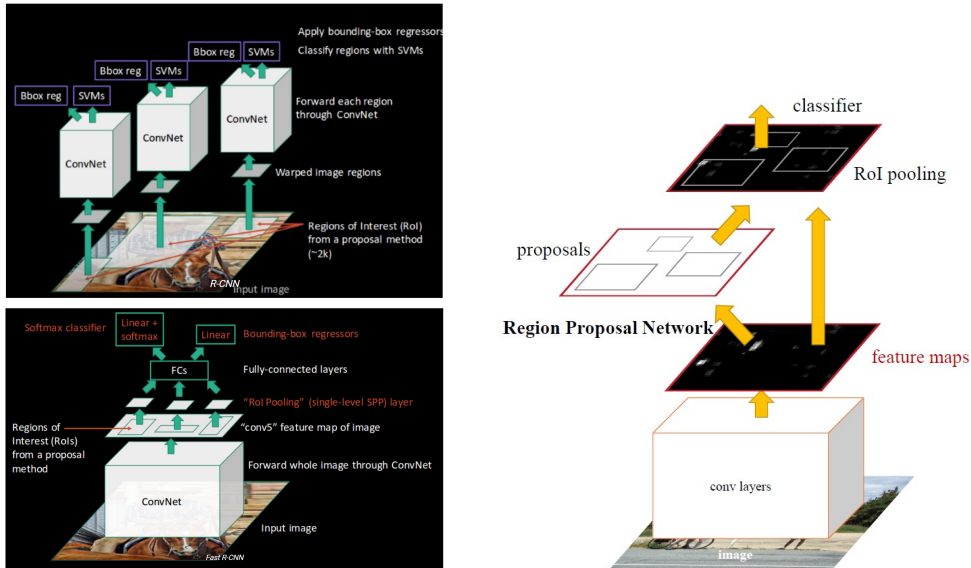


Figure 3: From <http://www.robots.ox.ac.uk/~tvg/publications/talks/fast-rcnn-slides.pdf>: model scheme for R-CNN (top left), Fast R-CNN (bottom left), Faster R-CNN (right).

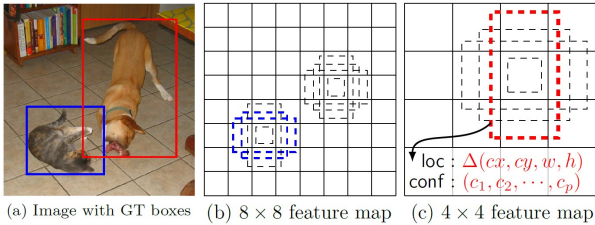


Figure 4: From [11], SSD framework. Out of the many default boxes, two are matched with the cat and one with the dog, which are treated as positives and the rest as negatives

boxes are grouped into a single one (“non-maximum suppression”, [16]). Moreover, the high number of negatives leads to a significant imbalance between negative and positive samples for training, which is overcome by only using the ones with highest confidence loss (a part of the overall loss function, measuring how confident the network is of the “objectness” of the box), so that the ratio between the negatives and positives is at most 3:1 (“hard negative mining”).

4.2 YOLO (You Only Look Once)

As in SSD, YOLO [12, 13, 14] uses a single neural network for detection (Fig. 5).

The input image is divided into a grid of $S \times S$ cells, and B bounding boxes are produced for each cell. For each box, a score is calculated, indicating the confidence for that box to contain an object (of any class) and the accuracy of the box in terms of its IoU with the ground truth ($Pr(Object) \times IOU^{truth}_{pred}$). For each cell, C (C = number of possible classes) probabilities calculated, conditioned on the grid cell itself containing an object. At test time, these conditional class probabilities and individual box confidence predictions are multiplied, in order to obtain class-specific confidence scores for each box.

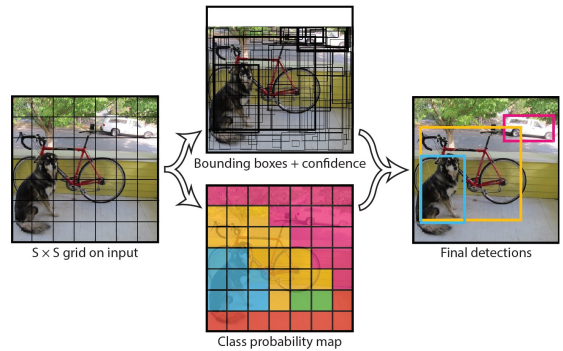


Figure 5: From [11], SSD framework. Out of the many default boxes, two are matched with the cat and one with the dog, which are treated as positives and the rest as negatives

5. COMPARISON

A performance comparison between the various presented algorithms, although being of great interest, can be tricky. Standard metrics like mAP [2] do not take into account factors like time and memory usage [7], which are of vital importance when real-time performance is required. On the other hand, greater speeds are obtained by sacrificing some accuracy, and it is important to be aware of the mechanisms influencing this trade-off. Finally, results reported by the various papers are generally obtained by using different settings, which makes their comparison less (if at all) significant. A work by Google research [7] offers a survey to study the trade-off between speed and accuracy for a series of models, including Faster R-CNN and SSD. The various presented models have been re-implemented in TensorFlow and trained on the MS COCO dataset. The effect of adopting different feature extraction architectures (MobileNet [6], VGG-16 [18], Inception, etc.) for each of the models has also been tested. In the following, we briefly summarize their main results.

- SSD models are faster on average, but cannot beat the Faster R-CNN in accuracy. Faster R-CNN requires at least 100 ms per image (Fig. 6), while SSD with MobileNet as feature extractor provides the best accuracy tradeoff within the fastest detectors. The highest accuracy is achieved by Faster R-CNN using Inception ResNet as feature extractor with 300 proposals, running at 1 second per image;
- Choice of feature extractors impacts detection accuracy for Faster R-CNN, but it is less important for SSD;
- For large objects, SSD can outperform Faster R-CNN in accuracy with faster extractors, but its accuracy drops significantly for smaller objects;
- Input image resolution strongly impacts performance: on average, reducing image size by half lowers accuracy by 15.88% and inference time by 27.4%;

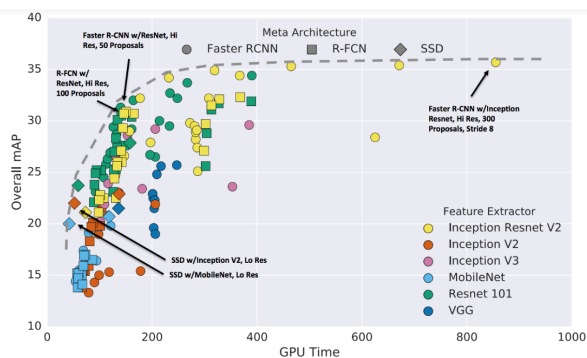


Figure 6: From [7], accuracy vs time. Different marker shapes indicate meta-architecture, while colors indicate feature extractor.

6. CONCLUSIONS

In this paper, we presented an overview of the region-based (R-CNN, Fast and Faster R-CNN) and of the single shot (SSD, YOLO) families of algorithms for object detection. An analysis of the speed and accuracy of the model parameters, and of how they are influenced by the choice of the models parameters, is briefly summarized.

7. ACKNOWLEDGMENTS

The research was done in the context of the “CoachMyLife” project, co-financed by the AAL Program (AAL-2018-5-120-CP) and the Ministry of Public Administration of Slovenia.

8. REFERENCES

- [1] D. Erhan, C. Szegedy, and A. Toshev. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2147–2154, 2014.
- [2] M. Everingham, L. Van Gool, C. K. Williams, et al. *International journal of computer vision*, 88(2):303–338, 2010.
- [3] R. Girshick. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.
- [4] R. Girshick, J. Donahue, T. Darrell, et al. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.
- [5] K. He, X. Zhang, S. Ren, et al. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [6] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. *arXiv preprint arXiv:1704.04861*, 2017.
- [7] J. Huang, V. Rathod, C. Sun, et al. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7310–7311, 2017.
- [8] A. Krizhevsky, I. Sutskever, and G. E. Hinton. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- [9] A. Kuznetsova, H. Rom, N. Alldrin, et al. *arXiv preprint arXiv:1811.00982*, 2018.
- [10] T.-Y. Lin, M. Maire, S. Belongie, et al. In *European conference on computer vision*, pages 740–755. Springer, 2014.
- [11] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.
- [12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.
- [13] J. Redmon and A. Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017.
- [14] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.
- [15] S. Ren, K. He, R. Girshick, et al. In *Advances in neural information processing systems*, pages 91–99, 2015.
- [16] R. Rothe, M. Guillaumin, and L. Van Gool. Non-maximum suppression for object detection by passing messages between windows. In *Asian conference on computer vision*, pages 290–306. Springer, 2014.
- [17] P. Sermanet, D. Eigen, X. Zhang, et al. *arXiv preprint arXiv:1312.6229*, 2013.
- [18] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [19] C. Szegedy, W. Liu, Y. Jia, et al. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.
- [20] C. Szegedy, A. Toshev, and D. Erhan. In *Advances in neural information processing systems*, pages 2553–2561, 2013.
- [21] A. Toshev and C. Szegedy. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1653–1660, 2014.
- [22] K. E. Van de Sande, J. R. Uijlings, et al. In *ICCV*, volume 1, page 7, 2011.
- [23] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola. 2019. <http://www.d2l.ai>.
- [24] K. Zhang, M. Sun, T. X. Han, et al. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(6):1303–1314, 2017.

EkoSmart asistent za iskanje po integracijski platformi

David Golob
IJS
Jamova 39, 1000 Ljubljana
+386 1 477 3147
david.golob@ijs.si

Matjaž Gams
IJS
Jamova 39, 1000 Ljubljana
matjaz.gams@ijs.si

Aleš Tavčar
IJS
Jamova 39, 1000 Ljubljana
ales.tavcar@ijs.si

Povzetek

V tem prispevku opišemo delovanje asistenta za iskanje po EkoSmart integracijski platformi. Namen asistenta je, da omogoči lažje in hitreje iskanje informacij po integracijski platformi. Asistent je še posebej priročen za starejše in manj večje uporabnike internetnih tehnologij. Asistent uporabi lematizator za določitev samostalnikov v uporabnikovem vprašanju. Asistent nato uporabi dobljene samostalnike za iskanje po integracijski platformi. Če asistent ne pridobi nobenih zadetkov za statistične podatke ali API-je, se obrne na starega EkoSmart asistenta za odgovor.

Ključne besede

Asistent, EkoSmart, integracijska platforma.

1. Uvod

Namen asistenta za iskanje po EkoSmart integracijski platformi, je da omogoči poenostavljeno iskanje po EkoSmart integracijski platformi, kjer so na voljo različni statistični podatki ter API-ji (npr. API za občinskega asistenta ter API za zdravstvene čakalne vrste).

Namen projekta EkoSmart je razviti ekosistem pametnega mesta z vsemi podpornimi mehanizmi, ki so potrebni za učinkovito, optimizirano in postopno integracijo posameznih področij v enovit in povezan sistem vrednostnih verig. Projekt se je začel 1. Avgusta 2016 ter končal 31. Julija 2019. Vrednost projekta ja 8,66 mio EUR. V projektu je sodelovalo 25 partnerjev.

Integracijska platforma je sestavljena iz treh delov: podatkov, produktov ter storitev. Kot rečeno, se z našim asistentom osredotočimo na iskanje po podatkih ter storitvah (API-jih). V času pisanja, so bili na integracijski platformi dostopni 4 nabori podatkov (z imeni »Transport«, »Slovenski kraji«, »Zdravje« in »Okolje«) ter 13 storitev (API-jev). Kot platforma za dostop do API-jev se uporablja odprto-kodna platforma **wso2** [1]. Le ta platforma omogoča objavljanje ter prenos API-jev. Prav tako platforma omogoča upravljanje z objavljenimi API-ji v smislu monetizacije API-jev (i.e. koliko API-jev za koliko denarja). Za podatke se uporablja odprto-kodna **ckan** [2] platforma, ki omogoča objavi ter prenos podatkov.

Uporabnik se pogovarja z asistentom preko uporabniškega vmesnika, ki je dostopen na [3]. Uporabnik v uporabniški vmesnik vnese stavek/vprašanja o stvari, ki ga zanima (npr. »zanima me slovenija«) nato asistenta s pomočjo razumevanja naravnega jezika obdela uporabnikov stavek/vprašanje, pridobi stvar zanimanja ter izvede iskanje po naboru podatkov ter API-jev ter vrne primerne zadetke. Asistent, deluje tako, da iz uporabnikovega stavka, s pomočjo lematizatorja, izlušči samostalnike ter jih pretvori v osnovno obliko. Tako dobljene lematizirane samostalnike nato uporabimo za iskanje po bazi podatkov ter API-jev. Za iskanje po bazi podatkov že imamo na voljo storitev, ki nam sama vrne ustrezne zadetke. Za uporabo te storitve moramo samo priskrbeti iskane besede, za le te uporabimo lematizirane samostalnike. Za iskanje po bazi API-jev nimamo na voljo iskalne storitve, zato moramo iskanje implementirati sami. Vsak API ima svoje oznake

(angl. tags), ki služijo kot ključne besede. Princip iskanja po bazi API-jev je ta, da za API lematiziramo samostalnike v oznakah (kot to naredimo za uporabnikov stavek/vprašanje) ter nato primerjamo lematizirane samostalnike v oznakah z lematiziranimi samostalniki v uporabnikovem vprašanju ter si zabeležimo API kot zadetek v primeru ujemanja.

Uporaba asistentov/chatbotov-ov se je v zadnjem času z napredkom procesiranja naravnega jezika zelo povečala. Primer uporabe asistenta za avtomatsko sklicevanje sestankov je opisan v [4]. V tem članku je opisan asistent za sklicevanje sestankov preko elektronske pošte. Asistent iz naslova ter vsebine elektronske pošte, s pomočjo ključnih besed, identificira namen elektronske pošte. Če je namen sklic sestanka, asistent prebere celotno vsebino pošte ter s pomočjo tehnologije iskanja vzorcev izlušči ure ter kraj sestanka. Nato asistent preveri v koledarju če je sestanek mogoč, glede na to pošlje odgovor s podatki o možnosti sestanka. Z nekaj izmenjavami elektronske pošte, je asistent možen potrditi sestanek.

Podoben asistent za klepet, je opisan v [5]. Ta asistent je namenjen za namen servisiranja strank. In sicer, asistent dobi vprašanje/zahtevo od stranke, z uporabo procesiranja naravnega jezika je sposoben izluščiti namen in ključne besede povezane z namenom, s katerimi nato naredi iskanje po bazi znanja ter nato vrne odgovor stranki.

Prav tako je v [6] opisan sistem za tvorjenje asistenta za interakcijo s spletno stranjo. Asistenta uporabnik vodi/sprašuje z uporabo kazalca (na zaslonu) ter z govorom. Sistem je sposoben pretvoriti govor v tekst, ki ga nato pošlje asistentu. Sistem pregleda spletno stran, ko se le ta naloži, ter vstavi bazo znanja iz smiselnih semantičnih struktur (e.g. tabel, grafov, slik). Ko asistent sprejme tekstovni ukaz, pridobi namen ter entitete z uporabo agenta Watson. Z namenom ter entitetami nato pregleda bazo znanja ter ponudi odgovor. V primeru, da v bazi znanja ne odkrije zelenih podatkov, asistent prosi uporabnika za bolj podrobne napotke (npr. lahko ga prosi, da z kazalnikom pokaže na stvar zanimanja e.g. stolpec v tabeli).

2. Mehanizem delovanja asistenta

Lematizator, uporabljen v asistentu, je narejen posebej za Slovenski ter Srbo-Hrvaški jezik in je prosto dostopen na [7].

2.1 Iskanje po bazi podatkov

Za iskanje po bazi podatkov (od integracijske platforme) se uporabi platformo, ki je dostopna na naslovu <http://wso2.lavbic.net:5000/sl/>. Preko te platforme nato iščemo zadetke za lematizirane samostalnike, npr. če želimo rezultate za samostalnik "Slovenija", uporabimo zahtevo http://wso2.lavbic.net:5000/api/3/action/package_search?q=slovenija, ki nam vrne odgovor v JSON formatu, ki vsebuje podatke o zadetkih iskanja. Za vsak zadetek iskanja (o podatkih), si nato shranimo

- Url naslov, kjer so podatki dostopni.
- Naslov podatkov.

- Opis podatkov.

API za iskanje po bazi podatkov http://wso2.lavbic.net:5000/api/3/action/package_search?q=, išče po oznakah podatkov ter po opisu podatkov, tako vrne vse zadetke, ki ustrezajo posredovanemu konceptu/besedi. Npr. če nas zanima ozračje lahko pošljemo zahtevek http://wso2.lavbic.net:5000/api/3/action/package_search?q=ozra%C4%8Dje ter dobimo JSON odgovor, del tega je predstavljen v Slika 1. Najbolj pomemben del JSON odgovora je »resources« (slo. viri), ki je v delu »results« (slo. rezultati). Pod viri imamo ime podatkov (»name«), opis podatkov (»description«) ter spletni naslov podatkov (»url«).

```
"help": "http://wso2.lavbic.net:5000/api/3/action/help_show?name=package_s",
"success": true,
"result": {
  "count": 1,
  "sort": "score desc, metadata_modified desc",
  "facets": {},
  "results": [
    {
      "license_title": "Drugo (javna domena)",
      "maintainer": "Statisti\u010dni urad RS",
      "relationships_as_object": [],
      "private": false,
      "maintainer_email": "",
      "num_tags": 2,
      "id": "e2cfec8f-0c32-44e4-9abb-01c23ed57e17",
      "metadata_created": "2018-09-10T09:14:45.014710",
      "metadata_modified": "2018-09-10T09:19:28.235189",
      "author": "",
      "author_email": "",
      "state": "active",
      "version": "",
      "creator_user_id": "1afa4ee7-387d-454b-8d1c-dc2ce1b95dce",
      "type": "dataset",
      "resources": [
        {
          "mimetype": null,
          "cache_url": null,
          "hash": "",
          "description": "NAMEA emisije v zrak (SKD 2008) , Slovenija, letno",
          "name": "Koli\u010dina izpustov v ozra\u010dje",
          "format": "CSV",
          "url": "https://cloud.lavbic.net/s/bK06FywmRwMHQa7/download",
          "cache_last_updated": null,
          "package_id": "e2cfec8f-0c32-44e4-9abb-01c23ed57e17",
          "created": "2018-09-10T09:19:28.085789",
          "state": "active",
          "mimetype_inner": null
        }
      ]
    }
  ]
}
```

Slika 1:Primer dela API odgovora za iskanje po podatkih

2.2 Iskanje po bazi API-jev

Za iskanje po bazi podatkov o API-jih nimamo vgrajenega iskalnika v iskalni platformi (<http://wso2.lavbic.net:5000/sl/>). Zato pridobimo nabor API-jev z zahtevo <https://wso2.lavbic.net:9443/api/am/store/v0.11/apis>, ki nam vrne odgovor v JSON obliki (poglej sliko Slika 3 za primer odgovora). Ta odgovor vsebuje identifikacijski podatek (ID) za vsak API, z le tem podatkom pridobimo celotne podatke o dolo\u010denim API-jem; npr. \u0107e \u0177elimo pridobiti celotne podatke o API-ju z ID-jem **218466a1-ac3b-464d-ba21-f2605f8391e8**, pošljemo slede\u010do zahtevo <https://wso2.lavbic.net:9443/api/am/store/v0.11/apis/218466a1-ac3b-464d-ba21-f2605f8391e8>. Ta zahteva nam vrne odgovor v JSON format, v katerem so vsi podatki o izbranem API-ju. Za potrebe našega asistenta, si shranimo slede\u010de podatke:

- Ime API-ja.
- Opis API-ja.
- Oznake API-ja.
- Lematizirane oznake API-ja.

- Url naslov, kjer izbran API na voljo.

Za lematizacijo API oznak uporabimo isti lematizator, kot za lematizacijo uporabnikovega vprašanja/stavka. Prav tako iz vsake oznake (za izbran API), z lematizatorjem, izlu\u010dimo samostalnike ter jih pretvorimo v korensko/osnovno obliko. Tako obdelane oznake imenujemo lematizirane oznaka (kot v zgornjem seznamu).

Za primer API oznak izberemo API z ID-jem **0815aeb4-b191-48ca-aad5-244f12be701c** in imenom **24alife**.

Primer JSON odgovora za ta API je na sliki Slika 2. Pomembni deli JSON odgovora so: ime (»name«), opis (»description«), oznake (»tags«) ter spletni naslovi (»https« in »http«).

```
id: "0815aeb4-b191-48ca-aad5-244f12be701c",
name: "24alife",
description: "Storitev za pridobitev: najbolj pogostih aktivnosti i",
context: "/AlifePortal-Api/api/public/v1",
version: "v1",
provider: "ekosmart-ponudnik",
apiDefinition: "{\"swagger\":\"2.0\",\"paths\":{\"/top10Activities/{days}'",
aktivnosti za posamezno obdobje (danes - \u0161tevilo dni)\",\"x-auth-ty",
[{"name\":\"activityCode\",\"in\":\"path\",\"allowMultiple\":false,\"requirec",
aktivnost (koda aktivnosti, danes- \u0161tevilo dni)\",\"x-auth-type\":\"App",
[{"name\":\"days\",\"in\":\"path\",\"allowMultiple\":false,\"required\":true,\"",
pridobitev: najbolj pogostih aktivnosti za posamezno obdobje, najbc",
wsdlUri: null,
status: "PUBLISHED",
isDefaultVersion: true,
transport: [
  "http",
  "https"
],
tags: [
  "eZdravje",
  "24alife",
  "aktivnost"
],
tiers: [
  "Bronze"
],
thumbnailUrl: "/registry/resource/_system/governance/apimgt/applica",
endpointURLs: [
  - {
    - environmentURLs: {
      https: "https://wso2.lavbic.net:8243/AlifePortal-Api/a",
      http: "http://wso2.lavbic.net:8280/AlifePortal-Api/api",
    },
    environmentName: "Production and Sandbox",
    environmentType: "hybrid"
  }
],
businessInformation: {
  businessOwnerEmail: null,
  technicalOwnerEmail: "borut.radi@rc-ikts.si",
  technicalOwner: "RC IKTS \u0177alec d.o.o",
  businessOwner: "RC IKTS \u0177alec d.o.o"
}
}
```

Slika 2:Primer zahtevka za podatke o dolo\u010denem API-ju

```

count: 11,
next: "",
previous: "",
list: [
  {
    id: "0815aeb4-b191-48ca-aad5-244f12be701c",
    name: "24alife",
    description: "Storitev za pridobitev: najbolj pogostih aktivnosti za pos
context: "/AlifePortal-Api/api/public/v1",
version: "v1",
provider: "ekosmart-ponudnik",
status: "PUBLISHED"
  },
  {
    id: "5892db13-f93e-4ff6-a812-0da0a941e7a6",
    name: "CakalneVrste",
    description: "Storitev, ki omogoča posredovanje informacij o čakalnih do
context: "/cakalneVrste/v1.0.0",
version: "v1.0.0",
provider: "ekosmart-ponudnik",
status: "PUBLISHED"
  },
  {
    id: "6f13ab1c-e386-49f4-a8db-d1da07f91ea1",
    name: "cos_test-zgtva",
    description: "Storitev za pridobitev: najbolj pogostih aktivnosti za pos
context: "/cos_test-zgtva/v1.0.0",
version: "v1.0.0",
provider: "ekosmart-ponudnik",
status: "PUBLISHED"
  }
]

```

Slika 3: Primer odgovora na zahtevek za podatke o vseh API-jih

Ta API ima oznake (angl. tags) »eZdravje«, »24alife« in »aktivnost«. Te oznake služijo kot ključne besede, ki opišejo API. Kot rečeno, se oznake s pomočjo lematizatorja pretvorijo v osnovno obliko. V tem primeru so vse oznake že v osnovni obliki, tako da so lematizirane oznake »eZdravje«, »24alife« in »aktivnost«. V drugem primeru si ogledamo API z ID-jem 5892db13-f93e-4ff6-a812-0da0a941e7a6, imenom »CakalneVrste« ter oznakami »posegi«, »ezdravje«, »zdravnik«, »čakalne«, »zdravstvene ustanove«. V oznakah obdržimo le samostalnike v osnovni obliki. V našem primeru je oznaka »čakalne« pridevnik, zato se te oznake ne upošteva. Prav tako se oznaka »zdravstvene ustanove« (v tej oznaki je samostalnik samo »ustanove«) s pomočjo lematizatorja pretvori v »ustanova«. Tako nam lematizator za ta primer vrne sledeče lematizirane oznake: »poseg«, »ezdravje«, »zdravnik«, »ustanova«.

Podatke o API-jih zaradi časovne zahtevnosti ni smiselno shranjevati vsakič, ko uporabnik vnese vprašanje/stavek, v ta namen uporabimo "caching", kar pomeni, da si vsakič, ko je spletna stran z uporabniškim vmesnikom ([3]) naložena, shranimo podatke o API-jih v podatkovno bazo, iz katere nato beremo API podatke.

Iskanje po API podatkih (ki jih shranimo ob naložitvi uporabniškega vmesnika) poteka po sledečem postopku. Vsak lematiziran samostalnik (iz uporabnikovega vprašanja/stavka) primerjamo z lematiziranimi API oznakami (za vsak shranjen API), primerjava je narejena eksaktno (obe besedi se morata ujemati), za vsako ujemanje si shranimo ujemajoč API. Na tak način naredimo iskanje po API podatkih.

Z postopkoma za iskanje po statističnih podatkih ter API-jih pridobimo podatke, ki jih nato v določeni meri prikažemo uporabniku. Za statistične podatke prikažemo največ štiri zadetke (za vse samostalnike v uporabnikovem vprašanju/stavku skupaj), prav tako za API-je prikažemo največ štiri zadetke (za vse samostalnike v uporabnikovem vprašanju/stavku skupaj). Štirje zadetki, ki so prikazani so prvi zadetki, ki so bili najdeni z iskanjem po bazi podatkov ter API-jev. Npr. če so v uporabniškem stavku trije samostalniki, potem je vrstni red samostalnikov za iskanje po bazi podatkov ter API-jev kar vrstni red samostalnikov v stavku; npr. za stavek »Zanima me zdravje, slovenija in okolje«, je vrstni

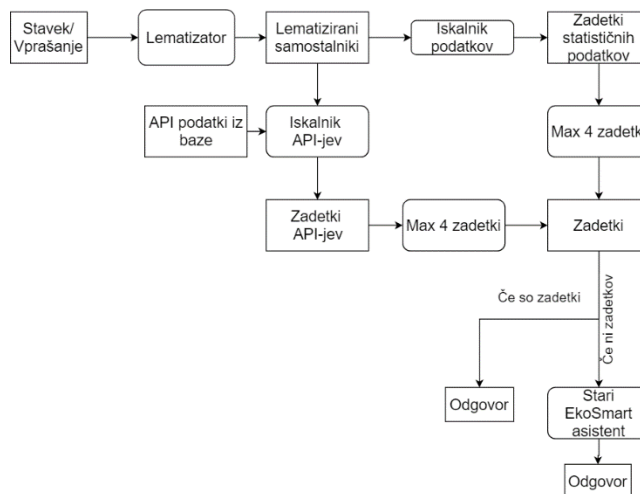
red iskanja po bati podatkov (ter API-jev) »zdravje«, »slovenija«, »okolje«. Za vsak samostalnik pridobimo določeno število zadetkov, na koncu pa prikažemo (največ) prvi štiri zadetke, ki so bili najdeni (za vse samostalnike skupaj). Tak mehanizem velja za iskanje po bazi podatkov ter bazi API-jev. Npr. da za primer »Zanima me zdravje, slovenija in okolje« predpostavimo, da ima (prvi) samostalnik »zdravje« 5 zadetkov, potem prikažemo prve 4 zadetke, ki so bili najdeni za »zdravje«, če pa npr. ima »zdravje« 2 zadetka ter »slovenija« 3 zadetke, potem prikažemo 2 zadetka za »zdravje« ter prva 2 zadetka za »slovenija« (le te dva zadetka sta prva zadetka, ki sta bila najdena v bazi podatkov oz. bazi API-jev). Vrstni red prikazanih zadetkov za posamezen samostalnik, je enak vrstnemu redu zadetkov v JSON odgovoru (za iskanje po bazi samostalnikov) oz. vrstnemu redu API-jev v bazi za iskanje po bazi API-jev.

2.3 Stari EkoSmart asistent

V primeru, da asistent ne najde nobenih zadetkov za statistične podatke in API-je, asistent posreduje uporabnikov stavek/vprašanje staremu EkoSmart asistentu. Poizvedba po starem EkoSmart asistentu je narejena s sledečo zahtevo <http://www.projekt-asistent.si/ekosmart/ask?question=>. Npr. če uporabnik vpraša/napíše "ekosmart" (in asistent ne najde nobenih zadetkov za statistične podatke ter API-je), dobimo odgovor starega EkoSmart asistenta s sledečim zahtevkom <http://www.projekt-asistent.si/ekosmart/ask?question=ekosmart>. Uporaba starega EkoSmart asistenta je uporabna za t.i. "small talk", ki ga le ta asistent razume.

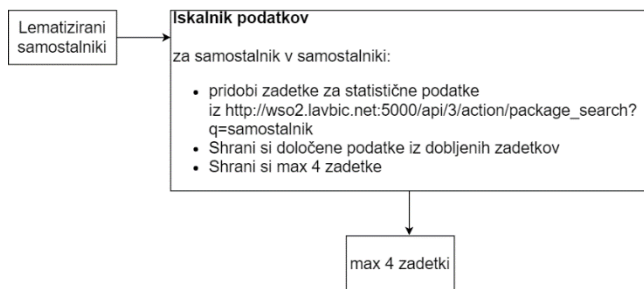
3. Shema

Na sliki Slika 4 je shema mehanizma delovanja asistenta.



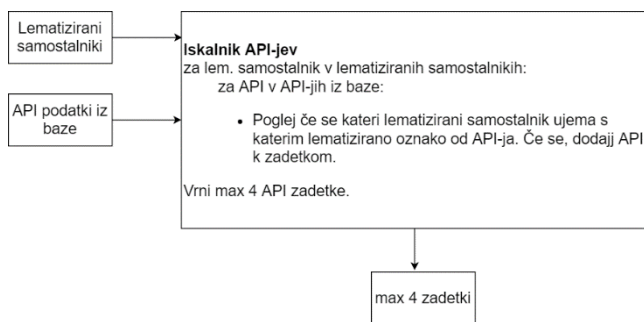
Slika 4: Shema delovanja EkoSmart asistenta za iskanje po integracijski platformi

Na sliki Slika 5 je ponazorjen mehanizem delovanja iskanja po bazi podatkov. Vhodni podatek v mehanizem so uporabnikovi lematizirani samostalniki (v osnovni obliki) izhodni podatek pa so maksimalno 4 zadetki.



Slika 5: Shema delovanja algoritma za iskanje po statističnih podatkih

Na sliki Slika 6 je opisan mehanizem za iskanje po bazi API-jev. Kot vhodna podatka sta uporabnikovi lematizirani samostalniki (v osnovni obliki) ter shranjeni API podatki iz baze, izhodni podatek pa so maksimalno 4 API zadetki.

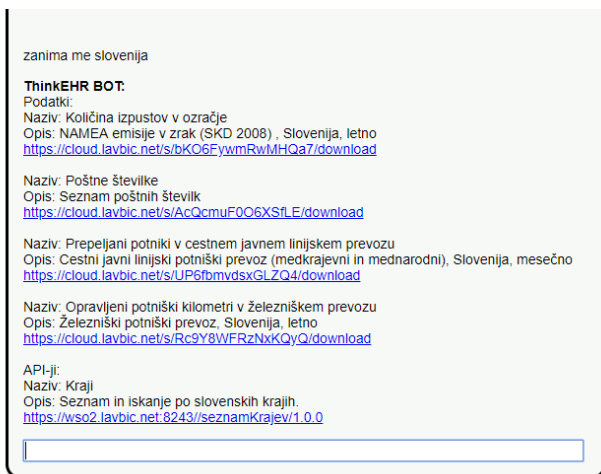


Slika 6: Shema delovanja algoritma za iskanje po API-jih

4. Rezultati in primeri.

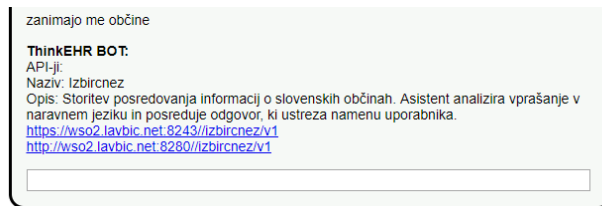
Prototipno rešitev smo namestili na spletno mesto [3]. Začetna inicializacija traja malce dalje (10-20 sekund) zaradi »prefetch« podatkov in njihovo pred procesiranje (kot je identifikacija samostalnikov ter njihova pretvorba v osnovno obliko). Nadaljnje poizvedbe se izvedejo brez časovnih zamikov.

Kot primer navedemo v uporabniškem vmesniku vprašamo "zanima me slovenija". Agent nam izpiše 4 zadetke za statistične podatke ter 1 zadetek za API-je.



Slika 7:Primer uporabe

Če npr. vprašamo "zanimajo me občine", dobimo kot odgovor 1 zadetek za API-je (izbirčnež).



Slika 8: Primer uporabe

Če pa v uporabniški vmesnik vnesemo "Živjo" (v tem primeru nimamo nobenih zadetkov za statistične podatke ter API-je, zato se asistent obrne na starega EkoSmart asistenta), nas agent pozdravi ter se predstavi.



Slika 9:Primer uporabe

5. Zaključek

V tem prispevku smo opisali delovanje EkoSmart asistenta za iskanje po integracijski platformi. Asistent uporablja robusten mehanizem za iskanje po bazi podatkov ter API-jev. In sicer asistent uporablja lematizator za identifikacijo samostalnikov in njihovo pretvorbo v osnovno obliko. Kljub malo daljši začetni inicializaciji poteka nadaljnja z asistentom nemoteno brez časovnih zamikov. Asistent je prestal različne preizkušnje uporabnikov ter se izkazal za robustna rešitev problema iskanja po integracijski platformi.

6. Literatura

- [1] [Elektronski]. Available: <https://wso2.com/>.
- [2] [Elektronski]. Available: <https://ckan.org/>.
- [3] IJS, „EkoSmart asistnet za iskanje po integracijski bazi,“ [Elektronski]. Available: <https://ekosmart.docker-e9.ijs.si/>.
- [4] N. Mhatre, K. Motani, M. Shah in S. Mali, „Donna Interactive Chat-bot acting as a Personal Assistant,“ *International Journal of Computer Applications*, pp. 6-11, 2016.
- [5] „Shallow Parsing Natural Language Processing Implementation for Intelligent Automatic Customer Service System,“ v *ICACSYS*, Jakarta, 2014.
- [6] M. Peveler, J. O. Kephart in H. Su, „Reagent: Converting Ordinary Webpages into Interactive Software Agents,“ v *Twenty-Eighth International Joint Conference on Artificial Intelligence*, Macao, 2019.
- [7] IJS, „Lematizator za slovenski in srbo-hrvaški jezik,“ [Elektronski]. Available: <https://repo.ijs.si/DIS-AGENTS/reldi-tagger>.

Virtual Assistant Aggregator for the Project Electronic and Mobile Health

Primož Kocuvan
Fakulteta za elektrotehniko
Tržaška cesta 25
1000, Ljubljana, Slovenia
primoz.kocuvan@outlook.com

Aleš Tavčar
Jožef Stefan Institute
Jamova cesta 39
1000, Ljubljana, Slovenia
ales.tavcar@ijs.si

Gregor Grasselli
Jožef Stefan Institute
Jamova cesta 39
1000, Ljubljana, Slovenia
gregor.grasselli@ijs.si

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

ABSTRACT

Electronic and mobile health (English: EMH, Slovene: EMZ) sub-project was part of the EcoSMART Smart Specialization program. During the EMZ sub-project, we developed several prototypes, products, and services that were integrated into a platform with the use of the Rocket.Chat application, which was developed as part of the AS-IT-IC project. Currently, ten entities and web services are integrated into the platform. Some of them like the municipality assistant consist of 200 single assistants. Others, like HEP-Y, an application for hepatitis, were designed before and represent a single entity. The EMZ platform simplifies the interaction with each of the services either by clicking or through natural language. When asked about healthcare, the platform searches through hundreds of possible replies and proposes the most promising.

Keywords

AS-IT-IC platform, Jaccard index, lemmatization, working agents, virtual assistants, Electronic and mobile health

1. INTRODUCTION

This paper describes a unifying platform addressing the specific needs of the healthcare domain and was developed as part of the EcoSMART S4 program [3]. It is based on the AS-IT-IC [6] platform which was developed in the cooperation of Austrian and Slovenian developers. AS-IT-IC, short for Austrian-Slovenian Tourist Information Center is a web application and a platform for assisting tourists who are traveling across Austria and Slovenia to plan a trip.

We have reorganized the internal structure of the source code and added new functionalities to meet the needs of the health care system. The EMZ platform implements several sub-applications, for example, ASPO [1] and HEP-Y [2], which are services for diagnosing and recognizing sexually transmitted diseases and hepatitis, respectively. The most important part of our platform is the ability to answer questions formulated in natural language by the user. The whole ecosystem is built and integrated into a popular and open-source application for messaging called Rocket.Chat.

During past projects, the Jožef Stefan Institute has already developed over 200 so-called chatbots or virtual assistants to be used on the web pages of the Institute and Slovenian municipalities. We integrated all of the existing chatbots into a single platform. Our platform is called in Slovenian language "Platforma za elektronsko in mobilno zdravje" or English "Platform for electronic and mobile health". The platform is for now in the Slovenian language, so it is intended for end-users who speak and understand Slovenian. The platform has three functionalities:

- It can answer questions posed in natural language.
- It can provide a link to the information we need.
- It can search through the database of Institute employees by applying lemmatization and stemming services to the search string and the data in the database.

In the following sections and subsections, we describe how the assistants, which are the building blocks of the platform, work, and which technologies we used to get the expected functionalities. Finally, we present some figures of the graphical interface and examples of conversations with the platform agent.

2. SYSTEM DESCRIPTION

Our top-level working agent in the EMZ platform works as follows:

- First, when a new question is received, the system processes the input. It asks all the working agents that are available about healthcare or the Eco-SMART project. These are the Summoner - Izbirčnej, the healthcare waiting queue assistant, the Eco-SMART assistant and the municipality assistants. Moreover, it also queries the database of prototypes and domains.
- If the question concerns assistants and not the database of prototypes and domains it sends the whole string by a POST or GET request to the online assistants. The assistants then perform lemmatization and stemming

of the text, search for the best answer, assign weights to all possible answers, and return the one with the highest weight.

- If the question concerns prototypes and domains, the system queries the database directly. However, before that, another set of lemmatization operations is applied. This process differs from the first since we integrated the search locally on the server where we read it from a file that is regularly updated.
- All the received answers are collected into a table. We filter the answers by selecting only those that the agent thinks are relevant to the initial query. Finally, we show them in the Rocket.Chat dialogue window.
- The user can select the most appropriate answer and start the conversation with the agent that provided the most appropriate response by clicking on one of the displayed responses. This functionality will be further described in the following text.

2.1 Assistans

Here we describe the assistants that provide possible answers to the requests in the EMZ platform. All are web applications and RESTful applications. REST is short for representational state transfer. This term means that we can communicate between different programs or processes by using HTTP requests. All of the agents are using JSON for the response format.

We will now provide a short description of the assistants.

2.1.1 Assistant for Queues in Healthcare ¹

The assistant identifies all medical institutions in Slovenia, where someone can apply for a medical procedure. It then displays the waiting time for the procedure in question at the institutions in a specific region, ordered by the waiting time from the shortest to the longest, to make it easier to identify where it is possible to get the procedure the fastest. The information about a medical facility includes contact information and the address. The system actively inquires the user for additional search constraints, such as the region in which we would like to receive the treatment and the urgency. It provides buttons that enable the user to see the available choices and to simplify communication.

2.1.2 Eco-SMART Assistant ²

The assistant can answer general questions about the Eco-SMART project. It responds to questions about the institutions and companies which are involved in the project. Next, it can provide organizational data from the project or information or services related to the execution and results of the project.

2.1.3 JSI Assistant ³

The JSI assistant called Robi is the first and most important assistant which was developed at the Institute. It's the fundamental building block for all other assistants. At first,

¹<https://df-chatbox.herokuapp.com/>

²<https://ekosmart.docker-e9.ijs.si/>

³<http://www.projekt-asistent.si/ijs>

it was implemented in Python 2.7, but we updated the code to Python 3.6. The main task was to delete all redundant code which was repeating. Now the assistant is slim and small, but we kept all functionalities from the older version. The assistant can answer questions about the people who are working on the Institute; it can open relevant web pages in the background for further searching. It has a multitude of available applications such as the Slovenian dictionary, a computer terminology dictionary and information about the menu at the canteen at the Institute. It also provides a quick way of informing the maintenance personnel of any problems requiring their attention.

2.1.4 Assistants for Municipalities in Slovenia ⁴

The Republic of Slovenia has 212 municipalities, and 200 municipalities have their assistant. It can answer general questions about the mayor, municipal council, environment and transport in the municipality and several other from the municipality related domain. The assistants are based on the original assistant Robi which works on the official web page of the institute. This system was integrated into the EMZ platform since it was agreed with the project partners.

2.2 Rocket.Chat

One of the reasons why we selected Rocket.Chat, as the underlying communication platform and for the integration of text messaging and communication with the assistant, is that it's a free and open-source project. The application provides hooks that make it easy to integrate a custom assistant. It makes it possible to send HTTP requests with the payload containing message inputs from the user, to a predefined address, and it displays the response to this request in the same chat room as the original inquiry. This mechanism was used to connect Rocket.Chat, which serves in the role of the front-end, with the developed back-end application written in Python using Flask.

Rocket.Chat also supports direct messaging, multiple rooms, and public channels. Since we are hosting Rocket.Chat ourselves, we have no limitations as we would if we were to use for example Slack, which limits the number of messages retained in history when using the free plan.

2.3 Lemmatization

Each text query to the platform is processed. First, it is split into individual words, producing a set of words. Second, we perform simple POS tagging on the text to identify nouns. This process is performed because nouns carry the most information when identifying the relevant entries in the prototypes and domain databases. Also, all punctuation is removed. Next, we apply lemmatization on the nouns and get a set of lemmas or dictionary from words. Then we calculate the Jaccard index on the lemmas, comparing words in the database to the words in the question string. The Jaccard index is described in more detail in the next subsection. Finally, we return the search result.

We have chosen lemmatization over stemming because it produced better results in our tests. Stemming only cuts off the extensions of the words. We need words in the dictionary

⁴<https://asistenti-website.docker-e9.ijs.si/post/seznam-asistentov/>

form. The downside of lemmatization is that it is slower and requires additional data in the form of a dictionary, which in our case was a model for Slovenian language, to apply morphological analysis. For this purpose we used a library that was also developed at JSI.

2.4 Jaccard Index

We can measure the similarity between two sets A and B with the Jaccard statistical formula (Equation 1).

First, we calculate the intersection between two sets and count the number of members. That number is our numerator. Then we calculate the union on the same two sets, count the number of members and put that number in the denominator. A result is always a number between 0 and 1 (as stated in Equation 2). If we multiply it with 100, we get a percentage. This method produces the Jaccard index, the higher the number we get, the more similar the two sets are. Complementary of the Jaccard index is the Jaccard distance. It measures how different two sets are.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (1)$$

$$0 \leq J(A, B) \leq 1 \quad (2)$$

If both sets are empty, then we consider the Jaccard index to be equal to one. We have applied the Jaccard index to the set of words obtained from lemmatization of the question string and the set of concepts stored in the database file. The higher the Jaccard index, the more similar we deem the question with the prototype or domain name. The system only returns the item with the highest score.

The Jaccard index is commonly used as an error measure when training convolutional neural networks to perform bounding box segmentation of images.

2.5 Blocking the Other Assistants When Talking to Just One

To prevent confusion when talking to multiple bots at the same time, we have implemented a function to focus the conversation to a selected assistant. This is achieved by tracing the value of the “talking” field associated with a particular assistant. Its value may be either True or False. When True then this assistant is also receiving questions, otherwise the working agent is “blocked” and does not participate in the conversation. The user sets the right value by clicking on the button to limit the conversation to a particular assistant. By clicking “Yes, I want to talk to the assistant” we set the “talking” field to True and for other assistants to False.

Besides allowing the user to have a more consistent conversation this also enables the users to select the conversation partner they believe will provide the right information even when the system is incapable of determining which answer to present correctly.

3. DEMONSTRATION OF THE PLATFORM

The platform works similarly as AS-IT-IC. It has two components, first the web page which provides information about projects, prototypes, and links to the relevant working agents. Second, the Rocket.Chat widget where we can search through the database or ask a question. Here we are talking to a bot, named EMZ-BOT. When we load the Rocket.Chat widget, we get an initial greeting from the bot (Figure 1). Here we can choose which working agent we want to talk to. If we don’t choose an agent, the EMZ-BOT provides answers from all working agents or assistants that responded.

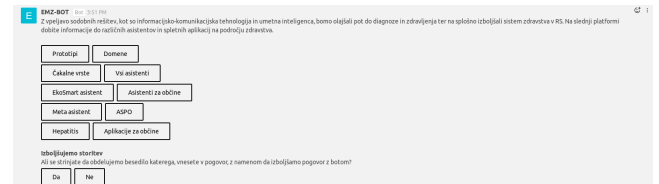


Figure 1: Example of an EMZ-BOT introductory greeting.

In the remainder of this section, we present 4 examples of interactions with the platform and communication with the different assistants.

We can ask the bot about the staff or people who work at the Institute. We receive an answer as shown in Figure 2.

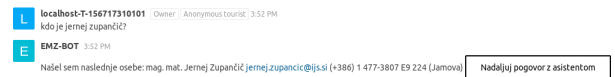


Figure 2: Example of an EMZ-BOT answer for looking up a person from JSI.

Next, if we have any medical problems, we can type it into the system. And the bot replies accordingly (see Figure 3).

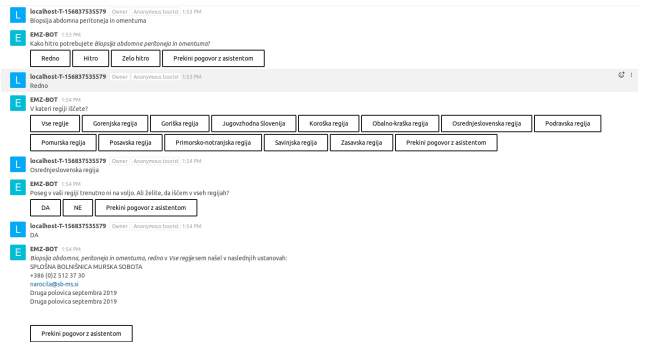


Figure 3: Example of a chat with EMZ-BOT when we need abdominal biopsy.

Also, we can search through the database of prototypes and domains. In Figure 4 we performed a search for heart failure. We get two answers, one from prototypes and one from domains or data. By comparing the answers, we can choose the assistant we want to talk to.

The last Figure 5 shows us the response from the bot when

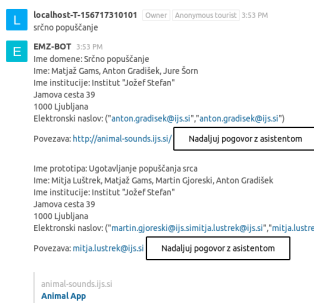


Figure 4: Example of an EMZ-BOT answer when we search for heart failure



Figure 5: Example of an EMZ-BOT answer when looking for something which the assistant does not find

it did not find anything after we have chosen to talk to just one working agent. We have an option to click on the button to talk to all agents.

4. FUTURE DEVELOPMENT OF THE EMZ PLATFORM

We have identified several issues and possible research directions for future work. The most important thing is to shorten the general response time of the platform. One possible solution would be to define functions in Python as coroutines and adding asynchronous behavior which is an abstraction of using threads. The authors of the Summoner [5] showed that using threads can shorten the response time by a factor of four. However, before speeding up the system, we should also run tests to evaluate the system’s ability to provide the correct answer. After this, we would be able to provide the assistants that are integrated into the system an improved and broader knowledgeable database. The assistants should learn from past questions and improve the database on the fly.

5. CONCLUSIONS

We have created a comprehensive platform using the latest information-communication technology based on the AS-IT-IC platform. One of the essential parts of this ecosystem is the EMZ-BOT. Users can communicate with it in natural language and obtain structured and informative answers. The system is intended to provide information and integrate services for the healthcare domain. Thus, it can be used by chronic patients and people with disabilities, as well as healthy individuals and the general population, who are just searching through the platform and seeking information about healthcare, information regarding municipalities, active life, the Eco-SMART project, and other. Soon, there will be a need for advanced ICT solution since our healthcare system is overwhelmed and in need of digitalization to support the need of patients. Our solution to these problems is the EMZ platform.

6. ACKNOWLEDGMENTS

The work and investment is co-financed by the Republic of Slovenia and the European Union from the Regional Development Fund. It is also co-financed by the Eco-SMART project. For help and support of the project we thank Jernej Zupančič and Miha Štravs.

7. REFERENCES

- [1] A. Ajanović, J. Konda, A. Peterlin, K. Počivavšek, G. Fele-Žorž, A. Gradišek, M. Gams, and M. Matičič. Application for sexually transmitted infection risk assessment. *Informatika 2017*, 41(2):253–254, June 2017.
- [2] A. Ajanović, A. Ulčar, A. Peterlin, K. Počivavšek, G. Fele-Žorž, A. Gradišek, M. Gams, and M. Matičič. Application for viral hepatitis infection risk assessment - hepy. *Informatika 2018*, 42(2):279–281, June 2018.
- [3] M. Drnovšek and M. Gams. Emz in ekosmart-asistent. *Information society 2018*, 1(3):11–14, September 2018.
- [4] M. Drnovšek, M. Gams, A. Tavčar, and G. Grasselli. Pregled asistentov ijs e9. *Information society 2018*, 1(3):15–19, September 2018.
- [5] A. Glavač, J. Zupančič, and M. Gams. The summoner - 'izbirčnež'. *Information society 2018*, 1(6):28–31, September 2018.
- [6] J. Zupančič and M. Gams. Austrian-slovenian intelligent tourist information center:project progress report 2018. *AS-IT-IC Workshop 2018*, E(21):5–9, September 2018.

Predictive Modelling of Feeling of Health for Congestive Heart Failure Patients

Jakob Valič
Jožef Stefan Institute
Jamova cesta 39
SI-1000 Ljubljana
jakob.valic@ijs.si

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

Borut Budna
Jožef Stefan Institute
Jamova cesta 39
SI-1000 Ljubljana
budna.borut@gmail.com

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

ABSTRACT

This paper focuses on predictive modelling of feeling of health (FOH) for congestive heart failure (CHF) patients. The basis for models is the data obtained from HeartMan project clinical trials, which involved 66 patients. The main indicator was FOH, reported repeatedly by patients. We used the Decision trees and Extreme gradient boosting algorithms to build the predictive models. In case of missing data, we used k-Nearest Neighbours imputation method. The algorithms predicted the FOH with around 70% accuracy. The relations in decision tree are in line with medical knowledge about CHF.

Keywords

HeartMan project, Feeling of health, Predictive modelling

1. INTRODUCTION

CHF is a serious disease, affecting 1-2% of population in western world. The percentage rises to more than 10% among people older than 70 years. [9] There is no cure available. However, if the disease happens, proper disease management is crucial as it can improve patient's quality of life (QoL).

The main actions to improve the health of CHF patients are quite well established: they include medications, physical exercises, mental exercises, proper nutrition, pill intake monitoring, etc. Modern technology enables monitoring more parameters related to health and QoL than was possible ever before. Furthermore, QoL does not depend just on physical health, but is subjective and can only be reported by the patients themselves. Patient-reported outcomes (PROs) are thus becoming increasingly important in medicine. [3] For example, Kang et al. [5] included telemonitoring parameters and PROs in a model predicting rehospitalisation of CHF patients, achieving the AUC of 0.59. The HeartMan project built upon a somewhat similar Chiron project [2], which aimed at developing a decision-support system to estimate the health risk of the patients. The Chiron study included 24 patients, who used wearable devices, and provided information on how they felt each day. [7]

This paper deals with predicting self-reported FOH from parameters collected with monitoring technology. The data used in this paper was collected during the HeartMan project. [1] The project developed the application for self-management of the CHF. The data collected was also used to build predictive models of FOH. The idea of predictive modelling is to enable the choice of actions for improving patient's FOH. The HeartMan models provide short-term advice and interventions in comparison to more common long-term ones. [4] Furthermore, the advice could be personalized.

2. HEARTMAN CLINICAL STUDY

HeartMan clinical study was carried out in two trials: the first took place in Belgium, another in Italy. In Belgium, 36 patients were involved, 12 of which were in control group and 24 in intervention group. In Italy, 30 patients were involved. 80% of the patients were male. The mean age of patients was 63 years. Most of the patients were placed in New York Heart Association (NYHA) functional class II, other in NYHA class III, meaning they had (slightly) limited physical abilities. Most of the patients had been diagnosed with CHF for more than 18 months. The intervention group patients used the HeartMan system from 3 to 6 months. They were given the RuuviTag environmental sensor to gather information on air humidity, temperature and pressure. They wore HeartMan watch to gather the PPG signal, skin temperature, galvanic skin response, cortisol level readings, heart rate and RR intervals. Each day they measured their weight and blood pressure. They were given smartphone with HeartMan application.

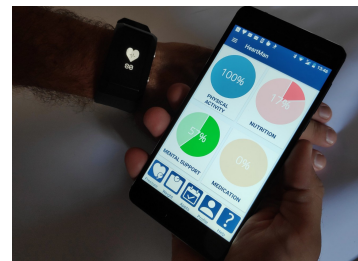


Figure 1: HeartMan watch and application.

HeartMan application gathered data from the HeartMan watch and sent it to the server. A crucial part of HeartMan application was the decision support system, which scheduled mental and physical exercises several times per week for each patient. On scheduled days, a notification was sent to the patient urging him to complete the exercise. Other than that, he could decide to do additional exercises. Each time a patient started one of the mental exercises, he was asked to answer a question: "How do you feel today?" He could then choose among the following answers:

- 1 = much better than usual
- 2 = somewhat better than usual
- 3 = about the same as usual
- 4 = somewhat worse than usual
- 5 = much worse than usual

The patient's answers regarding their FOH served as the input data for predictive models regarding the CHF patient's FOH. The distribution in Figure 2 shows us the number of FOH entries per patient from both Belgian and Italian trial. Some patients provided the information on their FOH every day, whereas some barely provided any.

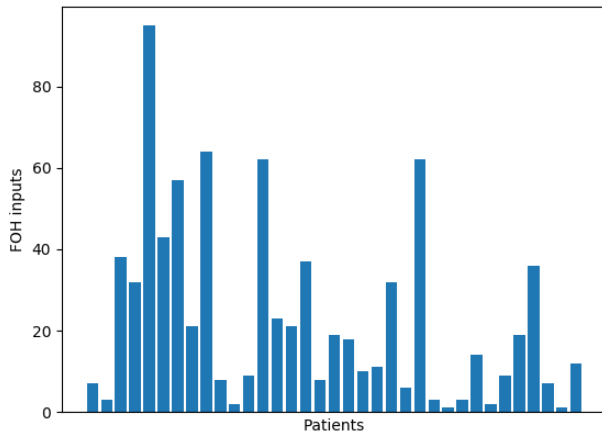


Figure 2: The distribution of FOH inputs per patient

3. PREDICTIVE MODELLING

3.1 Features

We used the following features collected by the HeartMan system:

- Air humidity, air temperature and air pressure, which were collected by RuuviTag sensor.
- Skin temperature, galvanic skin response and heart rate, which were collected by HeartMan wristband.

- Parameters from exercise report. The duration of the exercise, the type of exercise (endurance or resistance) and validity of exercise (check if pre-exercise requirements were met and the exercise was performed correctly) were generated by HeartMan application after exercise was performed by patient. The estimation of exercise intensity was input by patients.
- Systolic blood pressure and diastolic blood pressure, input by patients using HeartMan application.
- FOH, input by patients using HeartMan application.

Since the patients reported their FOH at most once per day, one instance for modelling corresponded to one day. We computed features describing the above parameters for each day. As a basis for calculation of features, we took the timestamp of patients' input on the FOH. Based on this timestamp, we calculated average and standard deviation of the parameters either for the same day as the FOH was recorded or for the previous day, or for last 3 or 24 hours. Table 1 shows the time intervals in which the various features were calculated.

Table 1: Dynamic features used for predictive modelling.

Parameters	Last 3 hours	The same day	Last 24 hours	Previous day
Skin temperature, Galvanic skin response, heart rate	✓	✗	✓	✗
Exercise parameters: duration, intensity, type, validity	✗	✓	✗	✓
Systolic blood pressure, diastolic blood pressure	✗	✓	✗	✓
Weight change since the previous day	✗	✓	✗	✗
FOH	✗	✓	✗	✗

We refer to the features in Table 1 as dynamic, because they were obtained daily via sensors. But we also considered static features from the patients' health records, which generally did not change during the trial, e.g. age, gender, body mass index, heart rate at rest, CHF etiology, patient's comorbidities, ergometry maximum load, ... In this paper we will focus on modelling dynamic features.

3.2 Modelling Methodology

We started the modelling with all the data of all the days when the patients reported their FOH. Then we had to address the issue of missing values: most features had the values for some of the days missing, and in many cases the missing values were quite numerous (because the patients did not use the HeartMan system fully on some days, or because of technical problems).

Missing values were filled in with imputation. The best imputation method on Chiron data proved to be k-Nearest Neighbours [8], which sets each missing value to the mean value of the same feature of k most similar instances. However, when a feature had a value missing on many days, imputation proved counterproductive, so we only used it on features with missing values on up to 30 days.

We then selected a number of thresholds for the maximum allowed number of missing values for a feature. We excluded all the features that had more missing values than each threshold, and we only included in further analysis the instances that had no missing values of the remaining features. Consequently, the stricter the threshold, the fewer features were left, and the more instances without missing values we could use.

The FOH was reported using the numerical representation, ranging from 1 (much better than usual) to 5 (much worse than usual). Like on the Chiron data, it proved too difficult to distinguish between all five classes, so we merged 1 and 2, as well as 4 and 5, ending up with three classes. Again, like on the Chiron data, even this proved challenging, so we removed the middle class in some experiments, reducing the prediction to only two classes.

Two machine learning algorithms were used to build predictive models: Sci-kit Learn implementation of Decision trees (DT) and Extreme gradient boosting (XGB). DT are human-understandable models that can be used to gain an insight into the relations in the data, whereas XGB models are ensembles of Decision trees that typically offer higher accuracy at the expense of understandability. We compared the results of these two algorithms to the majority model, which always returns the class with the most instances.

4. RESULTS

Our experiment consisted of building three- and two-class models on dynamic features. We compared them for various missing-value thresholds and both machine-learning algorithms. The results are shown in Table 2.

Table 2: Classification accuracy of prediction on dynamic features.

Dataset information					
Missing values	$<\infty$	<318	<205	<130	<50
Features	72	30	22	18	13
Instances – 3 classes	221	349	592	686	745
Instances – 2 classes	91	143	229	275	316
Results three classes					
Majority	0.59	0.59	0.62	0.61	0.61
DT	0.56	0.52	0.44	0.43	0.5
XGB	0.65	0.57	0.55	0.56	0.56
Results two classes					
Majority	0.52	0.52	0.53	0.51	0.52
DT	0.67	0.63	0.63	0.6	0.55
XGB	0.7	0.68	0.54	0.54	0.58

We can see that the results with two classes are not much better than those of the majority model, so we focused our experiments on two classes. To obtain reasonable accuracy, we needed enough features, so we had to accept features with many missing values. We find the most satisfactory results those with the missing-value threshold of 318 (in bold). If we exclude Italian data from our dataset, the XGB accuracy for a comparable missing-value threshold goes up to 0.76. This

is probably because of the higher number of missing values in the Italian data.

Figure 3 shows the DT with the missing-value threshold of 318. Labels value = [x, y] denote the number of instances when the patient felt good (x) vs. bad (y). Orange colour denotes feeling good and blue colour feeling bad. We can see that low systolic blood pressure generally means feeling bad, which makes sense in CHF patients who have problems with heart output. [6] The main exception is when the standard deviation of the heart rate is high, which is also reasonable, since high deviation means that the heart can adapt to varying demands. When the blood pressure is high, patients nevertheless feel bad when their average heart rate is high, which is also in line with expectation. Other parts of the tree make only minor contribution to the overall prediction.

Table 3: Classification accuracy of prediction for individual patients.

Personal data	None	10%	20%
Majority	0.43	0.42	0.42
XGB	0.58	0.62	0.85

Since the idea of the HeartMan project was to use predictive models to advise patients on how to improve their (feeling of) health, in our second experiment we tested models on individual patients. For each patient, we first built models on data of other patients, and then added 10% and 20% of that patient’s personal data. The results for two classes are shown in Table 3. We can see that person-independent models (personal data = None) did not perform well, although they still outperformed the majority classifier by 15 percentage points. Adding 10% or 20% of personal data to the training data for the models improved the accuracy substantially.

5. CONCLUSIONS

The general models built on HeartMan data proved reasonably accurate. A direct comparison with Chiron models is difficult because the majority classifier there had a much higher accuracy, but the results can be considered comparable. The relations in the models seem in line with the existing knowledge about CHF. Interestingly, the relations regarding ambient temperature and humidity, which were quite important in Chiron models, do not appear here.

The personalised models built on Chiron data were a disappointment, while they are fairly accurate in the case of HeartMan. It is difficult to say why this is the case. One possible reason is that the question about the FOH was such that the patients could answer more consistently: they were asked about their FOH compared to the usual one, whereas Chiron patients were asked about their FOH compared to the previous day. Regardless of the reason, these results are very encouraging, since building this type of models was a major objective of the HeartMan project. They certainly warrant further investigation, and are a strong argument for future research on predictive models in personal decision support systems and in health systems.

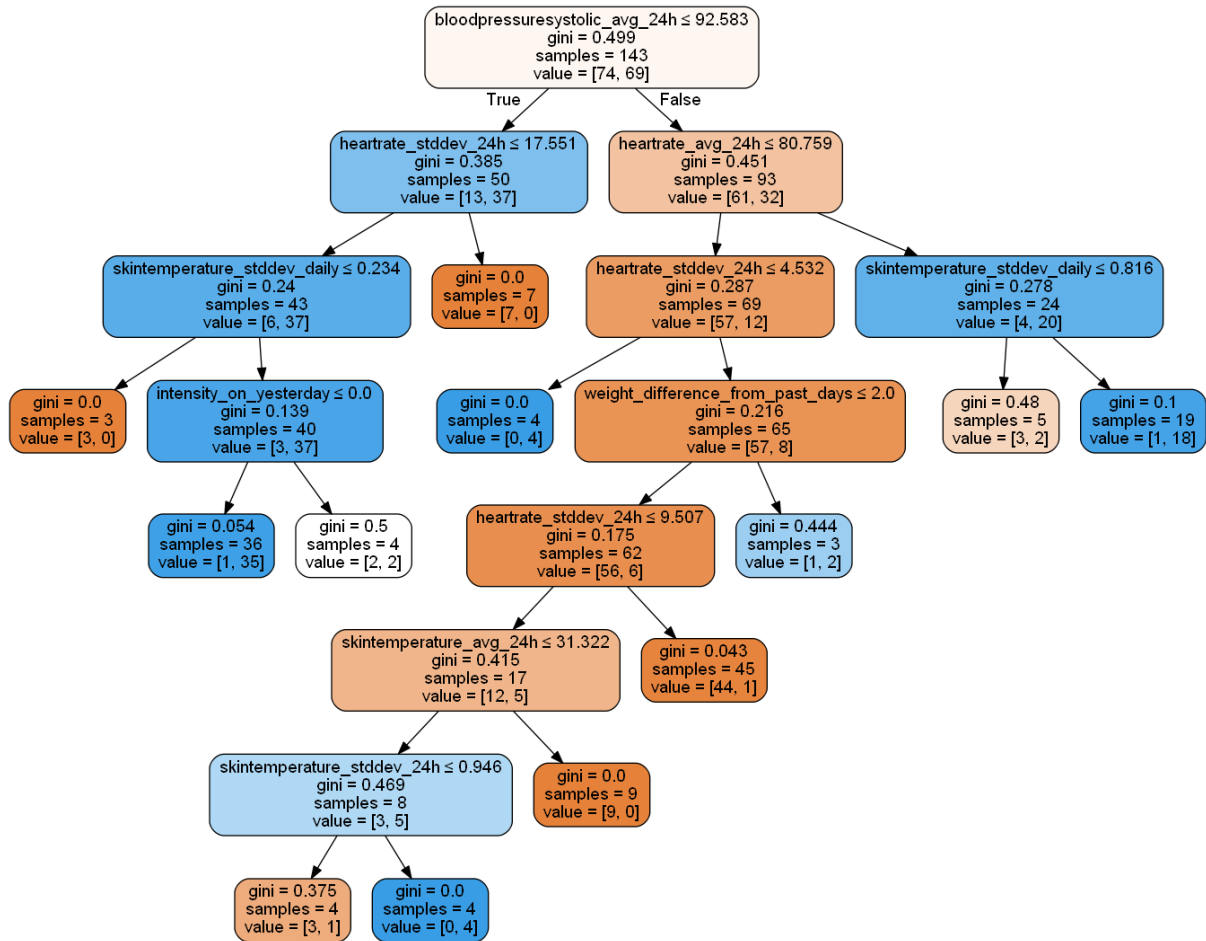


Figure 3: DT for two classes. Orange = feeling good and blue = feeling bad.

6. ACKNOWLEDGEMENT

The HeartMan project has received funding from the European Union’s Horizon 2020 research and innovation programme, grant agreement No 689660. Project partners are Jožef Stefan Institute, Sapienza University, Ghent University, National Research Council, ATOS Spain, SenLab, KU Leuven, Bittium and European Heart Network.

7. REFERENCES

- [1] Personal decision support system for heart failure management, 2019. <http://heartman-project.eu/>.
- [2] The Chiron project, 2019. <https://medrecord.io/project/chiron/>.
- [3] S. D. Anker, Agewall, et al. The importance of patient-reported outcomes: a call for their comprehensive integration in cardiovascular clinical trials. *European Heart Journal*, 35(30):2001–2009, July 2014.
- [4] A. Baert, D. D. Smedt, J. D. Sutter, et al. Factors associated with health-related quality of life in stable ambulatory congestive heart failure patients: Systematic review. *European Journal of Preventive Cardiology*, 25(5):472–481, 2018.
- [5] Y. Kang, M. McHugh, et al. Utilizing home healthcare electronic health records for telehomecare patients with heart failure. *Computers, Informatics, Nursing*, 34(4):175–182, 2016.
- [6] T. T. Lee, J. Chen, D. J. Cohen, and L. Tsao. The association between blood pressure and mortality in patients with heart failure. *American Heart Journal*, 151(1):76 – 83, 2006.
- [7] M. Mlakar, P. E. Puddu, M. Somrak, et al. Mining telemonitored physiological data and patient-reported outcomes of congestive heart failure patients. *PLOS ONE*, 13(3):1–23, March 2018.
- [8] M.-C. Monard. A study of k-nearest neighbour as an imputation method. October 2002.
- [9] P. Ponikowski, A. A. Voors, S. D. Anker, et al. 2016 ESC guidelines for the diagnosis and treatment of acute and chronic heart failure: the task force for the diagnosis and treatment of acute and chronic heart failure of the European Society of Cardiology (ESC). *European Heart Journal*, 37(27):2129–2200, May 2016.

Categorising Behavioural States of People with Profound Intellectual and Multiple Disabilities

Matej Cigale
Jožef Stefan Institute, Jožef
Stefan International
Postgraduate School
Jamova cesta 39
1000 Ljubljana
matej.cigale@ijs.si

Gašper Slapničar
Jožef Stefan Institute
Jamova cesta 39
1000 Ljubljana
gasper.slapnicar@ijs.si

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

ABSTRACT

People with profound intellectual and multiple disabilities (PIMD) are a diverse group of individuals. They face extreme difficulties in communicating to the outside world. This paper presents two specialised machine learning (ML) methods that attempt to classify the behavioural states and communication attempts of people with PIMD based on annotations of non-verbal signals (NVS) and expert knowledge. The first is based on the idea of unique NVS that classify the behavioural state, e.g., a smile in happy healthy individuals. The second uses the arousal-valence model as a scaffold to generate a value for valence based on a group of NVS.

Keywords

Behavioural state, Communication, PIMD, Prolog, Intent detection

1. INTRODUCTION

People with profound intellectual and multiple disabilities (PIMD) are a heterogeneous group, suffering from different ailments and conditions, making them face extreme difficulties in their daily lives. Severe cognitive, motor and sensory disabilities makes this population reliant on outside care for most daily tasks, and thus extremely vulnerable. While these individuals are exactly the ones that would benefit most from intelligent systems in their vicinity, they are unable to use them due to relative high complexity. The main problem is their lack of symbolic communication – they are unable to express their desires in a consistent manner.

Most assistive technology relies on some form of symbolic communication, which makes it unusable for people with PIMD. The INSENSION project aims to develop a system that will observe behavioural state and non-symbolic communication attempts of people with PIMD and interpret them to people in the vicinity in order to allow them to render assistance or support if needed; and even automatically control their environment using external services. This will in turn help the people with PIMD to have greater agency in their environment and help people who are not as familiar with them to assist them in meeting their daily needs.

The first step of the INSENSION system is to recognise non-verbal signals (NVS) expressed by people with PIMD (e.g., certain gestures [1] and facial expressions [7]) and important features of their environment (e.g., presence of a care-

giver and objects, temperature). Afterwards, these are interpreted as behavioural states and communication attempts, and provided to a caregiver or external services. This paper deals with the interpretation of NVS once they are recognised by the sensoric and machine vision sub-systems.

Interpretation of NVS of people with PIMD is a challenging task, since each individual is unique with different abilities and signals. Thus, no general-purpose system can be developed and personalised classification methods must be used. Mappings between certain NVS and behavioural states are known to those close to the specific person, and this expert knowledge should be used in the decision making process. Due to no possible generalization, we are also dealing with low amounts of data per subject, as collecting a large set of annotated data for each individual is neither practical nor feasible. Finally, as context is important for interpretation of behavioural states, a database of relevant contexts should be built.

This paper is organised as follows. In Section 2 we present the related work on the subject. Section 3 discusses the data collected so far. Section 4 presents the two ML methods: the Unique Non-Verbal Signals model optimised for extremely small data sets, and the Valence model that works better with limited but somewhat larger data sets. Section 5 looks at the results and discusses current and future work. Finally, Section 6 draws the conclusions based on this paper.

2. RELATED WORK

Since our focus is on recognising the ambiguous feelings, desires and intentions of people with PIMD, which are expressed in unique and unexpected ways, we focus on work dealing with detection and understanding of human feelings. Arousal and valence are the standard metrics that are used to map human feelings onto a 2D plane. Arousal can be understood as the strength of a feeling, while valence is the positive or negative connotation of the feeling. There are several ways to map discrete feelings to this 2D space and the actual mapping is not agreed upon, leading to some ambiguity on this subject, but it is at this point one of the standard models [6].

A step towards understanding feelings that is closest to what INSENSION will use (from video and audio) was done by Metallinou et al. [3]. They use a different space, also includ-

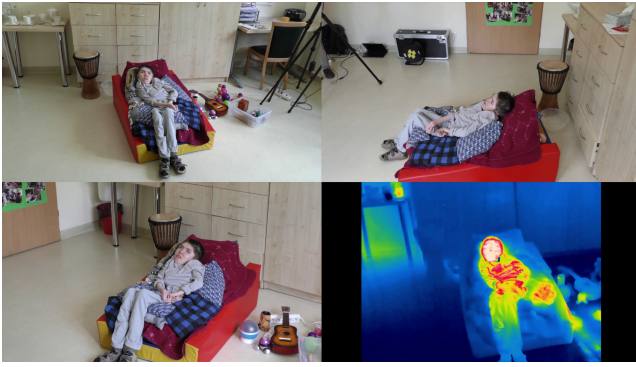


Figure 1: An example of the recorded videos. Computer vision algorithms are ran on all the streams and the results collated based on probability.

ing the dominance dimension. They used USC CreativeIT database consisting of acted-out scenes.

When it comes to extracting the context of the interaction, there are several approaches that produce interesting results. Probabilistic Event Calculus [4] is one of the approaches that can be used and extended to the case at hand. Case-based reasoning (CBR) [2] is another paradigm, in which knowledge is represented as a set of cases – events that happened, and the solutions that were used to solve the problem. Events that are detected are conformed into the closest case that is stored in the database and the solution of the problem is used. The solution is then evaluated and stored in the system based on the success of the solution.

3. DATA

Five PIMD people are currently involved in the INSENSION project. Expert knowledge was collected from their caregivers, who know them well, in the form of an extensive questionnaire. This data was then incorporated into the behavioural state recognition to improve decisions.

Visual data was collected in the facilities where the people with PIMD are cared for, and took the form of multiple-angle recordings with normal and thermal cameras (see Figure 1 for examples). Videos were annotated by hand, using the ELAN [5] software. Annotators were asked to input suitable pre-defined annotations and note any special cases that might play a role in the behavioural state of the subjects. Any state that was not specifically marked was considered *neutral*.

In our experiments the annotations of behavioural states were considered as ground truth, as we feel that people tasked with annotation were familiar enough with their subjects so that they could render as accurate picture of their behavioural state as is possible [8].

4. METHODOLOGY

Our methods assume that the person with PIMD has distinct NVS that correlate to his internal behavioural states. Each of the detected signals can have a meaning, but that is not guaranteed. The NVS can have no meaning or the same NVS can be used to convey multiple dissimilar meanings.

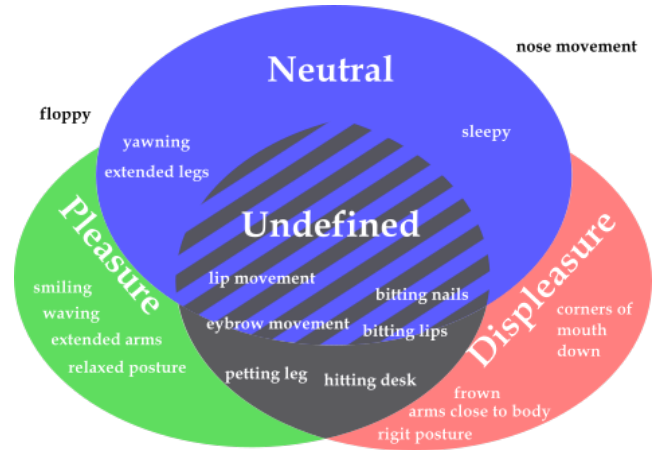


Figure 2: The visualisation of the NVS set interactions.

These signals do not necessary follow social conventions, for instance, lifting the corners of the mouth up can signify pain not pleasure as in normative individuals.

4.1 Classical Machine Learning

Deep architectures are not feasible in our case due to the small amount of available data. In our experiments we segmented the data into 3-second windows, owing to the fact that this was the window size used by annotators. Due to small number of examples leave one out approach was used and several training rounds were used to determine the accuracy of classification. Only present states were used in the classification.

Several methods were tested (nearest neighbors, linear SVM, RBF SVM, Gaussian process, decision tree, random forest, neural net, AdaBoost, naive Bayes, QDA), the decision trees providing the best results. The average classification accuracy for pleasure and displeasure is 63.8%. While the results do not seem bad at first glance, we can do better. We would like to make use of expert knowledge and perhaps even have access to the model and tweak it if the experts say that it does not make sense.

4.2 The Unique Non-Verbal Signals method

The Unique Non-Verbal Signals method encompasses the idea that there exists a NVS that will signify a specific behavioural state, but will never be used to signify any other behavioural state. This means that in order for us to robustly detect, for example, pleasure, we must remove all NVS that are associated with displeasure or neutral state. This leaves us with a set of NVS that uniquely represent the behavioural state of pleasure. Additionally, if experts annotated that a certain NVS corresponds to a behavioural state, we must take that into account. This approach is illustrated in Figure 2.

To decide the state we check if there are NVS specific to pleasure, either from the expert knowledge or from annotated examples (Listing 1). The term $window(Interval, NVS, -)$ unifies for any 3-second window. Term $assessment(State, NVS)$ will unify if experts noted that a NVS signifies a state.

```

decide_state(Interval, 'Pleasure') :-
    pleasure_marker(Pleasure),
    window(Interval, NVS, Annotation),
    member(NVS, Pleasure).

pleasure_marker(NVS) :-
    assessment('Pleasure', NVS).
pleasure_marker(NVS) :-
    window(Interval, NVS, _),
    window(Interval, 'Pleasure', _),
    not(displeasure_marker(NVS)),
    not(neutral_marker(NVS)).

```

Listing 1: Querying the behavioural state.

Failing that, *pleasure_marker* will check if there is a NVS in the annotations associated with pleasure and not with displeasure or neutral state.

4.3 The Valence method

The second method treats the significance of the NVS as an indicator of behavioural state on a continuous scale. We assume that each NVS has a certain correlation with valence. In our case valence is a number that is correlated with the three behavioural states (*displeasure*, *neutral*, *pleasure*), a simplified case of mapping feelings to an arousal-valence plane. Valence is assumed to be a value in $[-1, 1]$ interval, where displeasure is associated with negative and pleasure with positive numbers. If there is little or no correlation between pleasure and the expression, it should gravitate towards negative values, as shown in Listing 2. Inverse must be true for displeasure. *correlation_set(NVS, Behavioural_state, Num_correlations)* returns the number of all annotated intervals that contain a NVS at the same time as the behavioural state. The *intervals(Behavioural_state, Num_examples)* returns the number of all annotated intervals of a certain behavioural state.

```

valence(NVS, Valence) :-
    correlation_set(NVS, Pleasure, NVS_P),
    correlation_set(NVS, Displeasure, NVS_D),
    correlation_set(NVS, Neutral, NVS_N),
    intervals(Pleasure, P_Set),
    intervals(Displeasure, D_Set),
    intervals(Neutral, N_Set),
    Valence_direction is NVS_P/P_Set
    - NVS_D/D_Set,
    Valence_strength is 1 + P_Set
    + N_Set+D_Set,
    Valence is Valence_direction/Valence_strength.

```

Listing 2: The function that calculates the valence.

We determine the behavioural state based on the value of the sum of valence scores (Listing 3). The *calculate_valence* is a recursive function that sums the valence of a set of NVS, and returns 0 for an empty set.

The *P_Cut* and *D_Cut* variables determine the intervals of pleasure, displeasure or neutral behavioural state. We use constraint logic programming to determine the optimal values for these values based on the training set. At its core it is a minimisation problem where we try to find the thresholds for the intervals that produce the smallest classification error. The pseudo-code of the algorithm is presented in Listing 4. Here *count_errors(Bag, Errors)* is a function that returns

```

behaviour_state(NVS_Set, Decision, P_Cut, D_Cut) :-
    calculate_Valence(NVS_Set, Valence),
    (valence > P_Cut ->
     (Decision = Pleasure);
     (valence < D_Cut ->
     (Decision = Displeasure);
     (Decision = Neutral))).

calculate_valence([], 0).
calculate_valence([NVS | Rest], Valence) :-
    valence(NVS, V1),
    calculate_valence(Rest, V2),
    Valence is V1 + V2.

```

Listing 3: Determining the behavioural State.

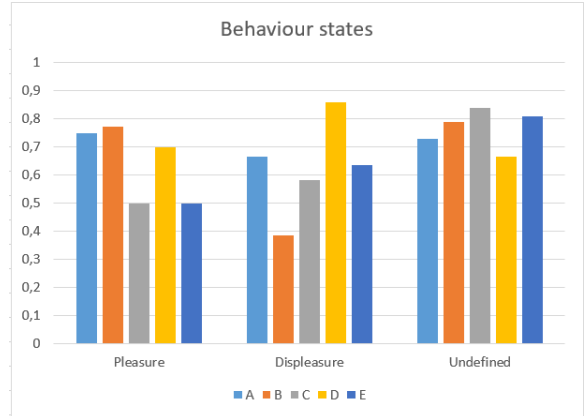


Figure 3: The accuracy from the decision tree classifier.

the number of missclassifications over the whole domain, and *min(Errors)* is a function that attempts to minimise the variable based on the constraints. It uses standard architecture for constraint logic programming on finite domains from the SWI-Prolog library, adapting it to the problem.

```

optimal_cut(P_cut, D_cut) :-
    [P_cut, D_cut] ins [inf..sup],
    min(Errors),
    findall(Correct,
            compare(Correct, P_cut, D_cut),
            Bag),
    count_errors(Bag, Errors).

```

```

compare(Correct, P, D) :-
    member(
    , All_observations)
    behaviour_state(NVS_Set, Decision, P, D),
    ground_Truth(NVS_Set, Truth),
    Correct = (Decision == Truth).

```

Listing 4: Calculating the minimal error.

5. RESULTS AND DISCUSSION

The benchmark we are trying to beat is based on standard machine learning algorithms from the python libraries. Out of the 10 algorithms tested, decision tree provided the highest accuracy (Figure 3).

The Unique Non-Verbal Signals method works surprisingly well with the limited data available. But it is expected to

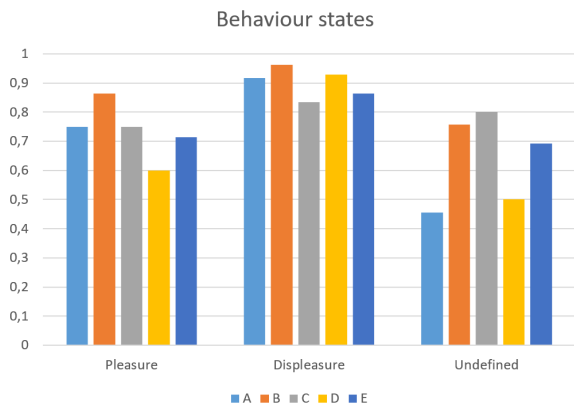


Figure 4: The Unique Non-Verbal Signals method classification accuracy.

become less viable with more data. The results in terms of classification accuracy are shown in Figure 4. The average classification accuracy for pleasure and displeasure is 81.8%. Due to uniqueness of the people with PIMD, a model is trained for each individual. The dataset for each individual was small, consisting of less than 10 annotated examples of each behavioural state. In order to evaluate our method, we trained the model on all the examples barring one for each state and compared it to the ground truth – based on annotations. The example here is a contiguous set of windows that annotate pleasure, so the data is not cross contaminated.

Using the same methodology, the Valence method performs worse than the somewhat naive Unique Non-Verbal Signals method, as seen in Figure 5. The average classification accuracy for pleasure and displeasure is 81.3%. Person A has very high miss-classification of neutral state, due to small sample size of this state. The Valence method seems to perform better for subjects with more annotations, perhaps indicating that it does not benefit as much from expert knowledge.

The two methods work on opposite spectrum, as Unique Non-Verbal Signals method with infinite data converges toward expert knowledge, while Valence method diverges from it.

6. CONCLUSIONS

In this paper we presented two ML methods specialised for learning behavioural states of people with PIMD. The advantage over the more common ML methods is the ability to incorporate prior knowledge in the form of assessments made by experts. The developed methods show promising results. The Unique Non-Verbal Signals method achieves 81.8% accuracy in classifying pleasure and displeasure and the Valence method achieves 81.3% on the same classes. However, more data must be collected and used to further validate our proposed methods. The next step is to use the environment of the person with PIMD to extract information on the actions that can be taken in order to ameliorate the state our user finds himself in.

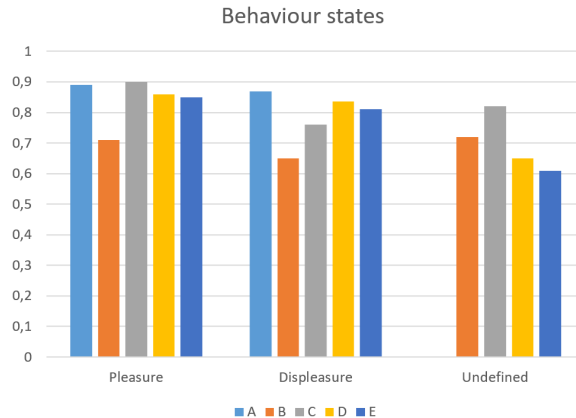


Figure 5: The Valence method classification accuracy.

7. ACKNOWLEDGMENTS

This work is part of the INSESION project that has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No. 780819.

8. REFERENCES

- [1] Z. Cao, T. Simon, S. E. Wei, and Y. Sheikh. Realtime multi-person 2D pose estimation using part affinity fields. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, (Xxx):1302–1310, 2017.
- [2] E. Lupiani, J. M. Juarez, J. Palma, and R. Marin. Knowledge Base Systems Monitoring elderly people at home with temporal Case-Based Reasoning. *Knowledge-Based Systems*, 134:116–134, 2017.
- [3] A. Metallinou, A. Katsamanis, and S. Narayanan. Tracking continuous emotional trends of participants during affective dyadic interactions using body language and speech information. *IMAVIS*, 31(2):137–152, 2013.
- [4] A. Skarlatidis, G. Paliouras, and A. Artikis. Probabilistic Event Calculus for Event Recognition. 2014.
- [5] H. Sloetjes and P. Wittenburg. Annotation by category - ELAN and ISO DCR. *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC'08)*, pages 816–820, 2008.
- [6] Z. Soroush, B. Brain, M. Z. Soroush, K. Maghooli, S. K. Setarehdan, and A. M. Nasrabadi. A novel approach to emotion recognition using local subset feature selection and modified Dempster-Shafer theory. *Behavioral and Brain Functions*, 4:1–15, 2018.
- [7] X. Sun, M. Lv, C. Quan, and F. Ren. Improved facial expression recognition method based on ROI deep convolutional neutral network. *2017 7th International Conference on Affective Computing and Intelligent Interaction, ACII 2017*, pages 256–261, 2018.
- [8] P. Vos. *Measuring the Subjective Well-Being of People with Severe and Profound Intellectual Disabilities*. PhD thesis.

Izdelava govorne zbirke za sintezo slovenskega govora

Development of a Speech Corpus for Slovenian Text-to-Speech Synthesis

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

Miro Romih
Amebis d.o.o.
Bakovnik 3
1000 Ljubljana
+386 1 831 10 35
miro.romih@amebis.si

Jerneja Žganec Gros
Alpineon d.o.o.
Ulica Iga Grudna 15
1000 Ljubljana
+386 1 423 94 40
jerneja.gros@alpineon.si

POVZETEK

V članku opisujemo potek izgradnje govorne zbirke za potrebe sinteze slovenskega govora. Zbirka bo primerna tako za HMM-sintezo, kot tudi sodobnejšo WaveNet-sintezo, preprosto pa jo bo prilagoditi tudi na korpusno sintezo. Opisujemo postopke za izbiro povi, izbiro govorcev, način snemanja govorne zbirke in njenega označevanja.

ABSTRACT

In this paper, we describe the process of building a speech corpus for Slovenian text-to-speech synthesis. The corpus will be suitable for both HMM synthesis and WaveNet synthesis, but it will also be easy to adapt to corpus-based synthesis. We describe the procedures for selecting text, speakers, and the process of recording and annotation a speech collection.

Ključne besede

Govorna zbirka, sinteza slovenskega govora.

Keywords

Speech Corpus, Text-to-Speech Synthesis for the Slovenian language.

1. UVOD

Za potrebe sinteze slovenskega govora v okviru projekta CityVOICE izdelujemo novo govorno zbirko z branih govorom. Takšen govor ustreza najpogostejšim oblikam rabe sintetizatorjev govora. Poleg tega je lažje izdelati njegovo transkripcijo, snemanje je bolj nadzorovano in predvidljivo. Pri spontanem govoru je govorno zbirko težko fonetično in prozodično uravnorežiti.

Najpomembnejši preostali dejavniki, ki smo jih upoštevali pri snovanju nove govorne zbirke za sintezo govora, so: izbira vsebine posnetkov, izbira govorcev, postopek snemanja in označevanje posnetkov.

Izbira velikosti govorne zbirke je posledica kompromisa med želenim številom variacij glasov oz. njihovim pokritjem na eni strani ter časom in stroški, vezanimi na razvoj, na drugi strani. Upoštevali smo tudi čas za kasnejše preiskovanje govorne zbirke in potreben prostor za njeno hranjenje.

Starejše obstoječe govorne zbirke so večinoma sestavljene iz posnetkov difonov, najpogostejših trifonov in nekaj besed. Obstoječa govorna zbirka eBralca [1] obsega 4.000 povi v obsegu nekaj ur posnetega in označenega gradiva. Nova zbirka bo predvidoma obsegala do 50.000 povi oz. do 100 ur posnetega gradiva. Za kvalitetno in naravno sintezo umetnega govora je zaželena čim obsežnejša in čim bolj raznolika govorna zbirka.

2. IZBIRA VSEBINE POSNETKOV

Umetno generirani govor mora zveneti naravno in biti prijeten za poslušanje. Pri izgradnji nove govorne zbirke smo na podlagi preteklih izkušenj [1,2] več pozornosti namenili:

- večji prozodični pestrosti posnetega besedila, ki pokriva najrazličnejše situacije rabe sintetizatorja govora, besedilo vsebuje tudi zelo kratke in zelo dolge povi,
- »pokrivanju« različnih prozodičnih kontekstov, v katerih se najpogostejše besede običajno pojavljajo,
- izogibanju besedam, ki niso vsebovane v slovarjih izgovarjav, s katerimi razpolaga projektna skupina, saj je zanje pravilen fonetični prepis potrebno izvesti ročno,
- vključevanju pogostih leksikalnih terminov oz. pogostih besed, kot so denimo telefonske številke, ekonomska terminologija, različne valute, terminologija s področja računalništva in interneta, medicine, pogosta lastna imena, nekatera tuja imena in izrazi, glavni in vrstni števnik, črkovanje ipd.,
- »pokrivanju« različnih situacij, ki nastopajo v dialogu (aplikacije dialoga in simultane prevajanja; npr. raba v virtualnih asistentih),
- večji zastopanosti raznovrstnih povi, predvsem vprašalnih in velelnih (pogostost teh povi je večja, kot je v samem besedilnem korpusu, iz katerega smo zajemali besedilo za branje),
- zajemanju različnih zvrsti novic, različnih napovedi (npr. vremenske napovedi), podajanju informacij (npr. stanje na cestah, borzne informacije) in navodil (npr. napotki za vožnjo),
- izbiri ustreznega ženskega glasu – ta je nekoliko nižji in bolj aspiriran (povprečna osnovna frekvenca je nižja kot pri aktualnem ženskem glasu),
- zagotavljanju enakih snemalnih pogojev med posameznimi sejami snemanja,
- obsegu govornega korpusa, ki je precej večji od obstoječega [1].

Izbir vsebine posnetkov oz. branih besedil govorne zbirke za sintezo govora:

- ustvarila se je obsežna tekstovna zbirka besedil, ki je pokrivala različne zvrsti (dnevni časopisi, revije, leposlovje ipd.); uporabili smo tudi besedilni korpus Gigafida, ki vsebuje 1,2 milijarde besed v slovenskem jeziku,
- tokenizacija – iz zbirke besedil smo odstranili vse oznake, vezane na oblikovno podobo (glava besedila, tabele ipd.),
- okrajšave, števila ipd. smo pretvorili v polno besedno obliko (normalizacija besedil),
- besedila smo pretvorili v predvideni fonetični prepis (grafemsko-fonemska pretvorba); izvedli smo ga z modulom za

grafemsko-fonemsko pretvorbo, s katerim razpolagamo projektni partnerji.

- obseg zbirke smo optimirali glede na vnaprej pripravljene kriterije (metoda požrešnega iskanja); pri tem smo si prizadevali zagotoviti statistično ustrezno vzorčenje izbranega področja govorjenega jezika.

Izbira povedi ni potekala naključno, pač pa je bila skrbno načrtovana. Postopek izbire povedi je potekal v več korakih:

1. Statistična obdelava besedil:

- Statistično obdelamo celoten besedni korpus in določimo pogostost pojavljanja posameznih glasov in glasovnih nizov v besedilu.
- Vključimo vse stavke (povedne, velelne, vprašalne itd.) in izdelamo statistiko posameznih vrst povedi oz. stavkov.

2. Izdelava spiska glasovnih nizov z oceno zaželenosti posameznega niza:

- V spisek vključimo nabor vseh teoretično možnih kombinacij difonov.
- V spisek vključimo vse trifone, štirifone in (po potrebi) ostale zaželene (najpogostejše) polifone, na katere smo naleteli pri statistični obdelavi besedil.
- Utež oz. ocena zaželenosti niza je odvisna od pogostosti njegovega pojavljanja v besedilu.

3. Postopek izbire povedi:

- Ocenimo doprinos glasovnih nizov za vsako poved iz tekstovnega korpusa.
- Doprinos povedi je enak vsoti vseh ocen zaželenosti nizov (iz spiska), ki se v povedi pojavijo.
- Doprinos posamezne povedi normiramo z dolžino povedi (št. besed v povedi ali št. fonemov v povedi).
- Določimo takšno utež, da bodo dolžine izbranih stavkov čim bolj ustrezale statistični porazdelitvi dolžin stavkov iz korpusa.
- Izberemo poved z najvišjim normiranim doprinosom.
- Iz spiska odstranimo vse glasovne nize, ki jih izbrana poved vsebuje.
- Ponovno ocenimo vsako poved in izberemo najboljšo (glede na novi spisek, v katerem so izločeni tisti glasovni nizi, ki smo jih že pokrili) ter popravimo spisek.
- Postopek ponavljamo, dokler ne izberemo želenega števila povedi.

4. Ovrrednotenje rezultatov:

- Vsakih 1000 povedi izdelamo statistiko difonov, trifonov, štirifonov in drugih polifonov, ki jih že pokrivamo (gre za glasovne nize, ki smo jih do takrat že izločili iz zgoraj omenjenega spiska).

5. Dodatne izboljšave algoritma:

- Ker mora zbirka vsebovati vse možne kombinacije difonov, algoritem popravimo tako, da difone dodatno utežimo glede na ostale polifone. Na takšen način algoritem na začetku daje prednost povedim, ki pokrijejo čim več novih difonov. Predvidoma se vsi difoni pokrijejo že po ca. 100 stavkih.
- Pri trifonih in štirifonih upoštevamo pri robnih glasovih tudi podatek o glasovni skupini, ki ji pripadajo (npr. štirifon "krak" ne bo doprinesel prav dosti novega v našo zbirko, če ta že vsebuje štirifon "krat"; zato oceno koristnosti takega štirifona popravimo navzdol). To lahko naredimo preprosto tako, da v spisek vnesemo dodatne nize skupaj z njihovimi frekvencami pojavljanja v korpusu (primer takega štirifona: "k"+"r"+"a"+"pripornik").

- Algoritem z različnim uteževanjem izboljšamo tako, da končni nabor vsebuje različne povedi (povedne, vprašalne, velelne, enostavne, sestavljene, naštevanje itd.). Tako lahko isti korpus učinkovito uporabimo tudi za generiranje prozodičnih parametrov pri sintezi govora.

3. IZBIRA GOVORCEV

Pridobili smo posnetke preko 10 različnih govorcev. Te krajše posnetke (nekaj deset stavkov z dobrim pokritjem difonov) smo nato strojno označili in preizkusili na aktualnem sintetizatorju govora. Posnetke je poslušalo več poslušalcev, ki so podali svojo oceno glede naravnosti in razumljivosti govora, pa tudi glede subjektivne ocene, kateri glas se jim je zdel najprijetnejši za poslušanje. V praksi se izkaže, da so nekateri glasovi preprosto bolj primerni za izdelavo sintetizatorjev govora kot drugi. Pri tem je težko vnaprej napovedati, ali je nek glas primeren ali ne, pri tem ni splošno veljavnih pravil.

Pri izbiri govorca smo upoštevali tudi njegovo sposobnost sledenja napotkom, potrebne ponovitve med snemanjem, čas snemanja ipd. Posneli smo en moški in en ženski glas. Smotno je, da sintetizator govora razpolaga s po dvema glasovoma za vsak spol.

Kandidate smo vnaprej seznanili z namenom snemanja in možne uporabe tako pridobljenih glasov. Pred snemanjem so morali izbrani govorniki podpisati pogodbo oz. privolilo, da dovolijo rabo posnetkov za potrebe sinteze govora.

4. POSTOPEK SNEMANJA

Snemanje govornega gradiva je potekalo ob prisotnosti izkušenega snemalnega operaterja z namenom, da se je preprečilo neustrezne izgovarjave besed in napake pri snemanju govora. Govorniku smo pred snemalnimi sejami podali ustrezna navodila in ga zaprosili, da povedi prebira razločno in enakomerno hitro. Med branjem besedila so imeli govorniki nameščene elektrode laringografa, s katerimi smo spremljali nihanje njihovih glasilk zaradi lažjega kasnejšega označevanja osnovnih period govornega signala.

Govor smo snemali preko kvalitetnega mikrofona v digitalni obliki in sicer na namenski računalnik v studiu. Pred mikrofonom se je nahajal ustrezen filter (angl. »anti-pop filter«), ki je zadržal razne pike, tleske ipd. Potrebna je bila še ustrezna mešalna miza, zaslon in slušalke (preko katerih govorec prejema navodila in poslušati povratni govor). Posnetke govora smo shranili v digitalni obliki na trajne računalniške pomnilniške medije. Frekvenca vzorčenja je 44,1 kHz, 24-bitno.

Samo snemanje celotne govorne zbirke je zaradi obsežnosti besedila, ki ga je bilo potrebno prebrati, trajalo več mesecev. Pri tem so morale nastavitve snemalne opreme ves čas ostati nespremenjene. Oseba, ki je snemanje nadzorovala, je preverila položaj govornika pred vsako snemalno sejo in jo primerjala s položaji v predhodnih sejah. Pred vsakim snemanjem je govorec poslušal svoje predhodne posnetke, s čimer se je skušal zagotoviti čim bolj enoten način govora med posameznimi snemalnimi sejami. Na začetku snemanja posamezne seje je govorec prebral nekaj vnaprej določenih fiksnih stavkov, ki so omogočali primerjavo glasnosti in višine govora med posameznimi snemalnimi sejami. Govorec ne sme biti preveč utrujen, zato je 10-minutno snemanje potekalo znotraj pol urnega intervala. Posamezna seja je trajala dve uri; znotraj tega časa smo lahko posneli 40 minut govornega materiala. Vsak govorec je opravil le eno dvournjo sejo na dan.

5. OZNAČEVANJE POSNETKOV

Uporabljamo tri nivoje anotacij oz. prepisov govornega besedila: grafemski prepis, fonetični prepis in prozodijske oznake. Ker je ročna segmentacija govora na fonetičnem nivoju naporna in dolgotrajna, smo pri tem uporabili vsaj delno avtomatizirane postopke, ki so bolj učinkoviti, če vnaprej poznamo grafemski prepis govornega gradiva. Avtomatskim metodam in postopkom je sledilo »ročno« popraviljanje oznak, kar je ne glede na hiter razvoj tehnologije še vedno zelo zamudno. Osnovne periode govornega signala smo označili s posebnim algoritmom temelječim na izhodnem signalu laringografa.

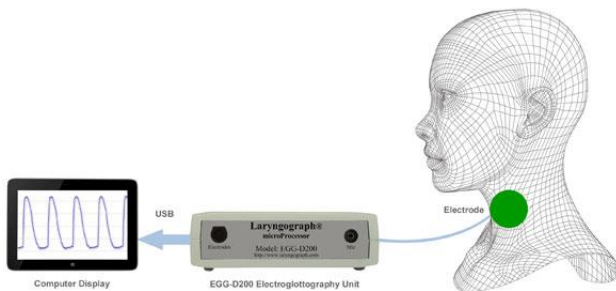
OPIS LARINGOGRAFA

Laringograf (slika 1) je elektronska naprava za nadzor in analizo vibriranja glasilk. Obsega strojno opremo (elektrode, mikrofon, laringograf mikroprocesor priključen na računalnik) in programsko opremo (Speech Studio). Akustični signal (govor) je posnet preko mikrofona, signal elektrolaringografa pa preko dveh elektrod nameščenih preko oz. na obeh straneh glasilk. Mikrofon in elektrode so priključene na laringograf mikroprocesor, slednji pa je priključen na računalnik (slika 2).

Deluje tako, da se preko dveh elektrod, nameščenih na obeh straneh govorne vrata na nivoju ščitnice, spusti šibak električni tok. Z napravo nato merimo električno impedanco vrata. Ker se impedanca spreminja z nihanjem glasilk, dobimo signal, ki je dobro koreliran z osnovnim tonom izgovorjenih glasov. Časovni potek impedance je označen kot signal Lx. Lx signal se lahko nadalje obdelava za namen pridobitve informacije o času trajanju period nihanja vokalnega trakta. Dobra lastnost signala laringografa je, da je odporen na akustični šum prisoten v signalu mikrofona [3, 4].



Slika 1: Laringograf z elektrodama [5].



Slika 2: Uporaba laringografa [6].

ZAJEM PODATKOV LARINGOGRAFA

Zajem podatkov laringografa obega pravilno izbiro elektrod glede na fiziološke značilnosti govornca, njihovo pravilno namestitvev in shranjevanje signala laringografa Lx.

Pravilna izbira velikosti in namestitvev elektrod laringografa je zelo pomembna. Za vsakega uporabnika laringografa si je bilo potrebno vzeti dovolj časa in prilagoditi nameščene elektrode tako, da je bil signal čim močnejši. Signal laringografa in njegovo jakost pri tem opazujemo na zaslonu računalnika. Signal je najmočnejši takrat, ko sta elektrodi na nivoju glasilk. Takrat je namreč električno polje med elektrodama najbolj podvrženo vplivu samega stika med glasilkama. Pri osebah z velikim izrazitim grlom je bila namestitvev elektrod preprostejša, saj smo lahko enostavno locirali obe strani ščitničnega hrustanca, dovolj velik kot elektrod pa je omogočal koncentracijo električnega polja okoli samih glasilk. Kadar grlo ni bilo tako izrazito, še posebej ko je bilo obloženo z večjimi količinami maščobnih blazinic, ga je bilo težje natančno locirati (položaj se med govornjem znatno spreminja). Posledično je bilo težje pridobiti »dober« in ustrezno močan signal laringografa Lx. Vendar pa za ugotavljanje osnovnih period k sreči zadostuje že relativno šibak signal. Detajli Lx signala pa so slabše razvidni oz. določljivi.

Pri ženski govorki (slika 3) smo preizkusili večje število elektrod. Kot primerne so se izkazale elektrode s premerom med 30 mm in 22 mm, medtem ko so se manjše elektrode (s premerom pod 16 mm) izkazale za neustrezne. Na koncu smo kot optimalni izbrali elektrodi s premerom 22 mm.



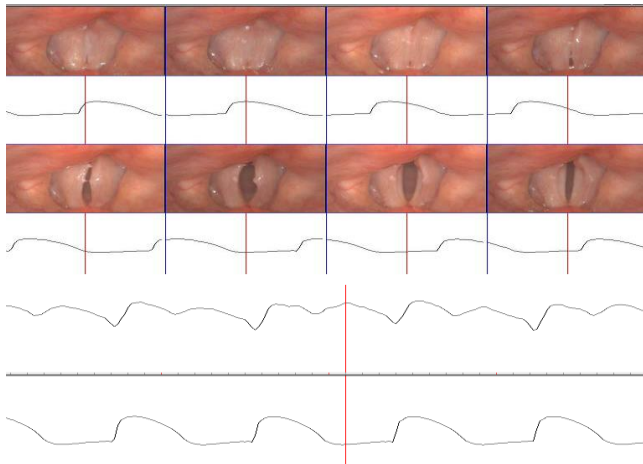
Slika 3: Govorka med snemanjem oz. branjem pripravljene gradiva v tonskem studiu. V ozadju tonski tehnik, ki med snemanjem na zaslonu spremlja tako signal laringografa Lx kot tudi mikrofonski signal Sp.

OBDELAVA PODATKOV LARINGOGRAFA

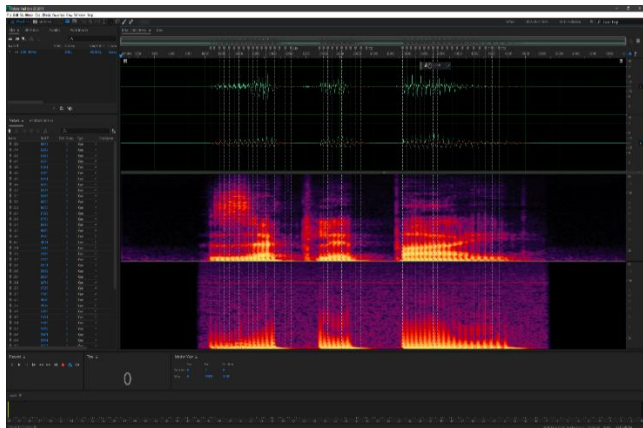
Obdelava podatkov laringografa obega označevanje osnovnih period govornega signala s pomočjo zajetega signala.

Slika 4 prikazuje osem zaporednih položajev glasilk med tvorjenjem zvonečnega glasu moškega govornca. Začetno zapiranje glasilk se odraža v hitro naraščajoči krivulji Lx signala. To zapiranje povzroči akustični odziv Sp (mikrofon) signala. Pri Sp signalu tipično opazimo krajši zamik glede na Lx signal. Najvišja točka signala Lx predstavlja trenutek, ko preko elektrod laringografa steče maksimalen tok. To ustreza trenutku, ko sta glasilki v največjem kontaktu oz. najbolj skupaj. Ko sta glasilki najbolj narazen, je amplituda signala Lx najnižja. Širina konice

signala pove, kako dolgo sta glasilki zaprti znotraj posameznega cikla vibriranja. Čas potreben za prehod iz ene točke signala L_x do enake točke na naslednji konici imenujemo perioda. Na takšen način lahko dobimo informacijo o osnovni frekvenci vibriranja glasilk (F_x). Informacijo o načinu odpiranja in zapiranja glasilk pa lahko dobimo iz naklona dviganja in spuščanja signala L_x . Prav tako lahko razberemo, kako dolgo sta bili glasilki zaprti.



Slika 4: Gibanje glasilk, moški glas, 120 Hz. Prikazana je serija osmih slik položaja glasilk, pridobljenih s stroboskopom, s pripadajočo oznako mesta na L_x signalu pod vsako sliko. Spodaj sta Sp (mikrofon) in L_x (laringograf) signala za šesto sliko. [5]



Slika 5: Primer govornega signala z označenimi osnovnimi periodami. Zgoraj je govorni signal posnet z mikrofonom, sledi signal laringografa L_x in spektralna prikaza obeh signalov. Navpične črte predstavljajo oznake period govornega signala.

Najprej smo preizkusili preprost algoritem za zaznavanje pulza in izračun časa trajanja med dvema impulzoma [7]. Začetek impulza predstavlja prvi vzorec, katerega amplituda je manjša od nič in hkrati manjša ali enaka amplitudi naslednjega vzorca. Konec impulza pa predstavlja zadnji vzorec, katerega amplituda je manjša kot nič in manjša ali enaka amplitudi predhodnega vzorca. Širina pulza je definirana kot časovna razlika med začetkom in koncem impulza. Vrh pulza predstavlja vzorec z največjo amplitudo med začetkom in koncem impulza (vrednost je vedno večja od nič). Trenutek pulza je določen kot prvi vzorec z amplitudo vrha pulza. Širina pulza je morala biti širša od od štirih vzorcev, vrh pulza pa je moral presegati neko arbitrarno določeno vrednost. Izračunal se je čas trajanja med dvema pulzoma, vrednost pa se je pretvorila v Hertze.

Potrebno je bilo izvesti še razločevanje med zvenečimi in nezvенеčimi deli signalov. V nadaljevanju pa smo uvedli še omejitve signala L_x in sicer tako, da je $L_x > 50$ Hz za moški glas in $L_x > 120$ Hz za ženski glas. Znotraj posameznega zvenečega predela pa so morali biti vsaj trije pulzi. Takšen algoritem je že uporabna referenca za določevanje osnovne frekvence govornega signala.

Zanesljivost prvotno zasnovanega algoritma smo izboljšali tako, da smo ga dopolnili z avtokorelacijsko metodo začetne ocene osnovne frekvence. Prav tako smo predlagali dodatne dopolnitve pri prepoznavanju osnovnih period zahtevnejših fonemov, kot je to npr. fonem »r«. Z analizo koeficientov HNM smo povečali natančnost določevanja osnovne frekvence ter sredine okna. Ob tem je bilo potrebno detektirati situacije, ko je osnovni algoritem »odpovedal« in takšne situacije še posebej obravnavati in razrešiti. Z obsežnim testiranjem smo prišli do končnega algoritma, ki se je izkazal kot dovolj robusten in natančen za potrebe izgradnje govorne zbirke za sintezo govora v okviru projekta CityVOICE.

6. ZAKLJUČEK

Pri izdelavi govorne zbirke CityVOICE smo posebno pozornost namenili določanju optimalnih pogojev za snemanje govornih zbirk, določanju optimalnih fonetično in drugače uravnoveženih vsebin za snemanje govornih zbirk ter rešitvam za iskanje optimalnih govorcev.

Pri označevanju osnovni govornih period smo si pomagali s signalom laringografa. Algoritem za označevanje osnovnih period prebranega besedila, skupaj s dodatno spremljevalno kodo, je bil implementiran v programskem orodju Matlab. Prvotno zasnovani algoritem je bilo tekom preizkušanja potrebno dopolniti z obravnavo bolj zahtevnih fonemov, kot je to denimo fonem »r«, poleg tega je bilo potrebno razviti še detekcijo in obravnavo posebnih situacij, ko je osnovni algoritem »zgrešil« pravilno namestitev oznak period.

7. ZAHVALA

Operacijo CityVOICE sofinancirata Republika Slovenija in Evropska unija iz Evropskega sklada za regionalni razvoj, in sicer v okviru »Operativnega programa za izvajanje evropske kohezijske politike v obdobju 2014-2020«.

8. LITERATURA IN VIRI

- [1] Žganec Gros, J., Vesnicer, B., Rozman, S., Holozan, P., Šef, T. 2016. Sintetizator govora za slovenščino eBralec. *Konferenca Jezikovne tehnologije in digitalna humanistika*, Ljubljana.
- [2] Šef, T., Romih, M. 2011. Zasnova govorne zbirke za sintetizator slovenskega govora Amebis Govorec, *Informacijska družba IS 2011*.
- [3] Wikipedia (<https://en.wikipedia.org/wiki/Electroglottograph>), 2019.
- [4] Laryngograph and Nasality processor, tehnična dokumentacija podjetja Laryngograph Ltd., 2019.
- [5] <http://www.laryngograph.com>, 2019.
- [6] <http://www.rose-medical.com/electroglottography.html>, 2019.
- [7] Bagshaw, P., C., Hiller, S., M., Jack, M., A., 1993. *Enhanced Pitch Tracking and the Processing of F0 contours for computer aided intonation teaching*. Center for Speech Technology Research, University of Edinburgh, UK.

Handling Real-World Problems within the COCO Platform

Tea Tušar
Jožef Stefan Institute
Ljubljana, Slovenia
tea.tusar@ijs.si

Dimo Brockhoff
Inria, CMAP, Ecole
Polytechnique
IP Paris, France
dimo.brockhoff@inria.fr

Vanessa Volz
modl.ai
Kopenhagen, Denmark
vanessa@modl.ai

Nikolaus Hansen
Inria, CMAP, Ecole
Polytechnique
IP Paris, France
nikolaus.hansen@inria.fr

ABSTRACT

Until recently, the problems employed for benchmarking optimization algorithms within the Comparing Continuous Optimizers (COCO) platform needed to have continuous variables and known optimal values. In addition, they had to be implemented within the platform (in the C language). These restrictions made COCO difficult to use for benchmarking algorithms on real-world problems. This paper describes the adaptations to the COCO platform that facilitate its use on real-world and other problems with integer or mixed-integer variables and unknown optimal values. Evaluation of solutions can now be done with external programs that are interfaced with COCO through socket communication.

Keywords

Real-world problems, algorithm benchmarking, the COCO platform

1. INTRODUCTION

Although Evolutionary Computation (EC) methods are often applied to real-world problems, they are almost exclusively benchmarked on artificial ones [7]. This is especially problematic in the field of Evolutionary Multi-Objective Optimization (EMO) where the most popular test problem suites like DTLZ [2] and WFG [5] have some unintended characteristics that stem from their construction and are not likely to be present in the real world. Consequently, we cannot expect algorithms that perform well on such test problems to also work well on real-world problems, which defies the very purpose of algorithm benchmarking [8].

To amend this issue, new test problems from the real world are being proposed. For example, the Mazda problem is a highly constrained problem with a large number of integer variables and two objectives [6]. It requires setting the thickness of several car parts so that their total weight is minimized and the number of parts with common thickness is maximized. The main challenge of this problem stems from its large search space dimension and the difficulty of finding feasible solutions due to the many constraints. Another example is the suite of three diverse design optimization problems that require Computational Fluid Dynamics (CFD) simulations for evaluating solutions [1]. These problems have a different number of objectives (two are single- and one is bi-objective) and can have varying search space

dimension. Since the CFD simulations are time-consuming, the biggest challenge is to find good solutions to the problems in reasonable time.

Using such problems for algorithm benchmarking is nontrivial since nothing is provided but the problems themselves. A researcher who wants to use these problems in a benchmarking study still needs to take care of the algorithm performance assessment as well as run additional algorithms on the same problems to acquire data for comparisons.

An alternative is to propose real-world problems within a framework that takes care of the cumbersome aspects of algorithm benchmarking. The Comparing Continuous Optimizers (COCO) platform¹ was designed exactly for facilitating the algorithm benchmarking task [3]. It incorporates several suites of test problems, takes care of all the performance assessment and makes it easy to include data from previous experiments in the comparisons. The selection of its problem suites was recently extended to include mixed-integer problems [9] as well as real-world problems based on games [12]. This required some adaptations of the platform that are also expected to simplify future inclusions of real-world problem suites.

This paper presents the modifications that were needed for COCO to support problems with integer variables, problems with unknown optimal values and external evaluation of solutions, which were not previously explained in [9] and [12]. While discussing the details of these changes, we provide some information that can be useful when adding additional suites of (real-world) problems into COCO.

After a brief presentation of the COCO platform and its latest suites in Section 2, we explain the adaptations needed to support real-world problems within COCO in Section 3. The paper ends with concluding remarks in Section 4.

2. THE COCO PLATFORM

2.1 Overview

The aim of the COCO platform [3] is to simplify the benchmarking of numerical optimization algorithms and make the data from those experiments available to the scientific com-

¹<https://github.com/numbbo/coco>

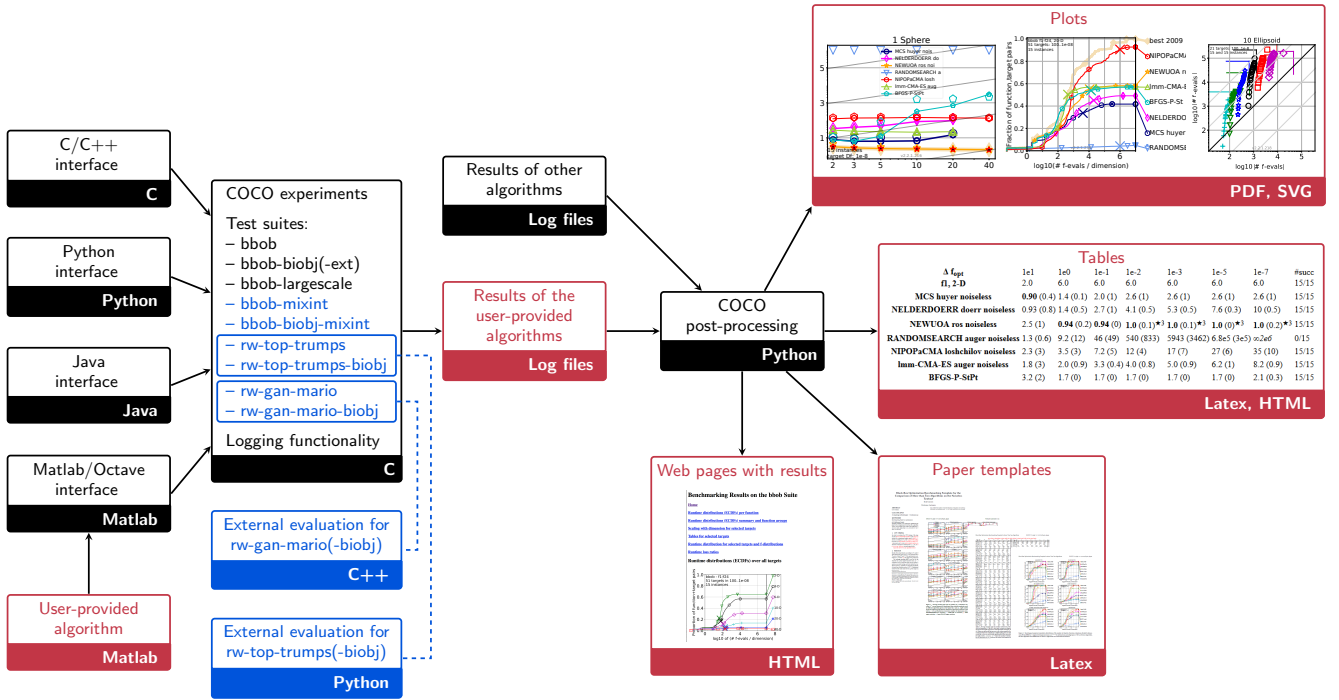


Figure 1: The COCO platform scheme. Its main components are presented in black, while the user-provided algorithm (shown here as implemented in Matlab) and its results are shown in red. Blue color is used to denote the recent additions.

community. The platform consists of two main parts (see Figure 1). The first, called *COCO experiments*, is implemented in C. It is used for running an algorithm on the chosen test problem suite and recording its performance. The algorithm can be connected to the platform using one of the available interfaces in C/C++, Python, Java and Matlab/Octave. During the algorithm run, its results are logged into files whenever one of the performance targets is achieved.

The second part, called *COCO post-processing*, is implemented in Python. From the log files created by the experiments, it produces plots and tables with information on the performance of the algorithm as well as HTML pages to facilitate browsing through them and paper templates with the most relevant results already included. With COCO post-processing it is very easy to add the performance of other algorithms to the comparisons. Currently, results of more than 300 experiments are available. Most were collected on the *bbob* suite of 24 continuous single-objective problems without constraints or noise [4].

Until of late, all the problem suites in COCO were based on the *bbob* problems. For example, the *bbob-largescale* suite contains large-scale instantiations of the *bbob* problems [11], while the *bbob-biobj* and *bbob-biobj-ext* suites consist of bi-objective problems constructed by using the *bbob* functions as their separate objectives [10].

2.2 Recent Problem Suites

We have recently proposed a total of six new problem suites that are more real-world-like than those previously included

in COCO [9, 12]. Table 1 shows summary information for some of their properties. All can be initialized with various search space dimensions and provide multiple instances that represent small perturbations of the problems. In all these suites the bi-objective problems were created by using two single-objective functions as the two objectives.

The *bbob-mixint* and *bbob-biobj-mixint* suites contain single- and bi-objective mixed-integer problems, respectively. They were constructed by discretizing the first 80% of the variables of the corresponding *bbob* and *bbob-biobj* problems. Since the discretization reduces the number of continuous variables, a large number of all variables need to be used in order to produce challenging problems. Therefore, the problem dimensions were set to be larger than those of the *bbob* problems, while the functions and instances remained the same.

The problems from the single- and bi-objective suites *rw-top-trumps* and *rw-top-trumps-biobj* are based on the Top Trumps card game. The goal (optimization problem) is to construct a deck for the game with desirable properties (objectives). The number of dimensions corresponds to the number of cards (22, 32, 42, 52) multiplied by the number of categories on a card (4), and the all-integer variable values are the values of the categories on the cards. Out of the five different objectives that measure a quality indicator of the deck, two can be computed directly and three require simulations of gameplay. The three bi-objective functions are constructed from the five single-objective ones in such a way that the two objectives are (at least partially) conflicting.

Table 1: Basic properties of the six recently proposed problem suites.

Suite name	bbob-mixint	bbob-biobj-mixint
# objectives	1	2
Dimensions	5, 10, 20, 40, 80, 160	5, 10, 20, 40, 80, 160
# functions	24	92
# instances	15	15

Suite name	rw-top-trumps	rw-top-trumps-biobj
# objectives	1	2
Dimensions	88, 128, 168, 208	88, 128, 168, 208
# functions	5	3
# instances	15	15

Suite name	rw-gan-mario	rw-gan-mario-biobj
# objectives	1	2
Dimensions	10, 20, 30, 40	10, 20, 30, 40
# functions	28	10
# instances	7	7

Lastly, the `rw-gan-mario` and `rw-gan-mario-biobj` suites contain single- and bi-objective problems of constructing levels for the well-known Super Mario Bros. platformer game to optimize the chosen objectives. The levels are generated by a Generative Adversarial Network (GAN), which learns a mapping from a continuous latent space with an arbitrary dimension to a valid level [13]. Properties of these generated levels (related to their difficulty and variety) are the optimization objectives. Out of the 28 single objectives, ten can be computed directly and the rest require simulations of gameplay. Again, the bi-objective functions were constructed by looking at the conflicts between objectives.

3. SUPPORTING PROPERTIES OF REAL-WORLD PROBLEMS

COCO was initially designed to work with the `bbob` problems that are continuous, have known optima and use the C code within COCO experiments to evaluate solutions. Here we explain in more detail the changes brought by the shift to real-world problems, which do not share these properties (see Figure 1).

3.1 Integer Variables

The Top Trumps and mixed-integer suites required supporting problems where either all or just some of the variables are integer. This entailed adding an additional parameter, which gives the number of integer variables to the internal problem class in COCO experiments as well as to the interfaces to all supported languages. Without any loss of generality we set that all the integer variables come before any continuous ones, which means that this single addition is enough to support problems with (some) integer variables (the parameter is naturally set to zero for continuous problems). The integer variables are internally still represented as real values with double precision. It is then up to the evaluation function to make sure they are correctly interpreted as integers.

In addition, the COCO loggers can be configured to output these variables as integers, which can save considerable space in case of a large number of integer variables (such as in the Top Trumps suites). This is done through the observer’s `log_discrete_as_int` parameter (set to false by default).

3.2 Unknown Optimal Values

In COCO, an evaluation is logged whenever it surpasses a target value. When an algorithm is run on problems with known optimal values, the target values are defined as differences to the optimal function value (in the single-objective case) or to the optimal value of a multi-objective performance indicator (in the multi-objective case). In the usual benchmarking setting in COCO, the targets are chosen equidistantly in logarithmic scale. Therefore, it is very important that the optimal value is known (or is at least very well approximated). If the estimate of the optimal value is (too) optimistic, the smallest target values will never be reached. If, on the other hand, the estimate is (too) pessimistic, the algorithm will be able to reach all targets while still being arbitrarily far away from the optimal value.

The discretization of the `bbob` and `bbob-biobj` problems that produced mixed-integer problems was performed in such a way that the optimal values remained equal and are therefore known (see [9] for more details). This means that similarly to their corresponding continuous predecessors, the optima for the `bbob-mixint` problems are known, while for the `bbob-biobj-mixint` problems, the ideal and nadir points are known, but not the Pareto sets and fronts (except for the special case of the double sphere function). In contrast, most Top Trumps and Mario GAN problems have unknown optimal values already in their single-objective formulation, which is to be expected in the majority of real-world problems. Consequently, neither the Pareto sets and fronts nor the ideal and nadir points are known for the bi-objective game-based problems.

While the issue of unknown optimal indicator values for the `bbob-biobj` problems is amended by providing an estimate of indicator values using all nondominated solutions from several runs of a number of algorithms, this approach is not feasible for real-world problems.

In order to support real-world problems with unknown optimal values, we are using an infinite number of equally spaced absolute target values aligned at zero with a step of 10^{-5} . This assumes that the difference between best and worst objective values is not much smaller than 1. In this way, the logger records an evaluation each time the algorithm finds a function (or performance indicator) value that improves the best found one by at least 10^{-5} . Such a strategy makes sure that the convergence to the optimal value can be detected (up to the precision of 10^{-5}) regardless of its absolute value.

After the experiments, the targets of interest need to be chosen for the post-processing part. This requires some preliminary analysis of the results. Once the targets are chosen, they can remain the same for future experiments or change in order to account for better solutions found later on. This does not affect the ability to add previously computed results to the comparison as the post-processing is always run anew.

3.3 External Evaluations

While artificial problem suites can be implemented in C with some moderate effort, this is much harder to do for real-world problems (especially those that are not originally available in C). To address this issue, we added the possibility to evaluate solutions using an external evaluator that is not provided by COCO.

This is achieved by the means of *socket communication*, where the external evaluator acts as a server waiting to be queried and COCO as the client that continuously queries the server with proposed solutions. In such a case, the ‘frame’ of the suite that provides the general information about its problems still needs to be implemented in COCO, however, this is rather straightforward and has been automated with a script.

Evaluation of solutions using socket communication works as follows. COCO (the client) sends to the external evaluator (the server) a solution together with the information needed to identify the problem, that is, the function and instance identifier and the number of dimensions. If needed, other parameters can also be passed at the same time. When the external evaluator receives the query, it evaluates the given solution with the right problem and returns the objective and constraint values as a response to the query.

This is a quite flexible and efficient way to communicate with an external evaluator. It is much faster than writing to and reading from files. It is also very versatile—the external evaluator can really be external (not even run on the same computer as COCO), which might be important for some real-world problems that cannot be disclosed.

4. CONCLUSIONS

By adding to COCO the support for problems with integer variables, unknown optimal values and external evaluation of solutions, we have opened its use for benchmarking optimization algorithms on real-world problems. We hope that the mixed-integer and game-based problem suites described in this paper are just the start and other real-world problems, such as the Mazda problem and the CFD problems mentioned in the Introduction will follow soon.

The code with the functionality described in the paper can be found at <https://github.com/ttusar/coco/tree/gbea>.

5. ACKNOWLEDGMENTS

The first author acknowledges the financial support from the Slovenian Research Agency (project No. Z2-8177).

6. REFERENCES

- [1] S. J. Daniels, A. A. M. Rahat, R. M. Everson, G. R. Tabor, and J. E. Fieldsend. A suite of computationally expensive shape optimisation problems using computational fluid dynamics. In *Proceedings of the 15th International Conference on Parallel Problem Solving from Nature, PPSN XV*, volume 11102 of *Lecture Notes in Computer Science*, pages 296–307. Springer, Cham, Switzerland, 2018.
- [2] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable test problems for evolutionary multiobjective optimization. In A. Abraham, L. Jain, and R. Goldberg, editors, *Evolutionary Multiobjective Optimization: Theoretical Advances and Applications*, pages 105–145. Springer London, 2005.
- [3] 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.
- [4] N. Hansen, S. Finck, R. Ros, and A. Auger. Real-parameter black-box optimization benchmarking 2009: Noiseless functions definitions. Technical Report RR-6829, INRIA, 2009.
- [5] S. Huband, P. Hingston, L. Barone, and R. L. While. A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation*, 10(5):477–506, 2006.
- [6] T. Kohira, H. Kemmotsu, A. Oyama, and T. Tatsukawa. Proposal of benchmark problem based on real-world car structure design optimization. In *Companion Proceedings of the Genetic and Evolutionary Computation Conference, GECCO ’18*, pages 183–184. ACM, 2018.
- [7] Z. Michalewicz. Quo vadis, evolutionary computation?: On a growing gap between theory and practice. In *Proceedings of the 2012 World Congress Conference on Advances in Computational Intelligence, WCCI’12*, pages 98–121. Springer-Verlag, 2012.
- [8] T. Tušar. On using real-world problems for benchmarking multiobjective optimization algorithms. In *Proceedings of the International Conference on High-Performance Optimization in Industry, HPOI 2018, 21st International Multiconference Information Society, IS 2018*, volume D, pages 7–10. Jožef Stefan Institute, 2018.
- [9] T. Tušar, D. Brockhoff, and N. Hansen. Mixed-integer benchmark problems for single- and bi-objective optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO ’19*, pages 718–726. ACM, 2019.
- [10] T. Tušar, D. Brockhoff, N. Hansen, and A. Auger. COCO: The bi-objective black-box optimization benchmarking (bbob-biobj) test suite. *ArXiv e-prints*, arXiv:1604.00359, 2016.
- [11] K. Varelas, A. Auger, D. Brockhoff, N. Hansen, O. A. ElHara, Y. Semet, R. Kassab, and F. Barbaresco. A comparative study of large-scale variants of CMA-ES. In *Proceedings of the International Conference on Parallel Problem Solving from Nature, PPSN XV*, volume 11101 of *Lecture Notes in Computer Science*, pages 3–15. Springer, 2018.
- [12] V. Volz, B. Naujoks, P. Kerschke, and T. Tušar. Single- and multi-objective game-benchmark for evolutionary algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO ’19*, pages 647–655. ACM, 2019.
- [13] V. Volz, J. Schrum, J. Liu, S. M. Lucas, A. Smith, and S. Risi. Evolving mario levels in the latent space of a deep convolutional generative adversarial network. In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO ’18*, pages 221–228. ACM Press, New York, 2018.

Risk Stratification of Cardiac Patients by Utilizing Additional Unlabeled Examples

Aleš Papič
University of Ljubljana
Faculty of Computer and Information Science
Večna pot 113, Ljubljana, Slovenia
ales.papic@fri.uni-lj.si

Zoran Bosnić
University of Ljubljana
Faculty of Computer and Information Science
Večna pot 113, Ljubljana, Slovenia
zoran.bosnic@fri.uni-lj.si

ABSTRACT

In the paper we address the challenge of cardiac patient risk stratification using the additional unlabeled data. The motivation for using unlabeled data comes from the field of semi-supervised learning (SSL), which has shown that additional unlabeled data can improve accuracy of supervised learning models. In addition to traditional SSL, we propose three new approaches that are based on active learning (AL), fuzzy learning (FL), and supervised clustering (SC). We evaluate them on the UCI ML heart disease dataset and with four different classification models. The results show that our approaches increase the inductive performance compared to the learning algorithms trained exclusively on labeled data. The most favorable performance was achieved with the fuzzy learning approach that utilizes a reliability estimate for selection of the most beneficial additional examples.

Keywords

risk stratification, cardiovascular diseases, machine learning, knowledge transfer, semi-supervised learning, active learning, fuzzy learning, supervised clustering, unlabeled examples

1. INTRODUCTION

Cardiovascular diseases (CVDs) are the leading cause of morbidity and death worldwide, together with cancer and chronic respiratory diseases. To prevent them, people with increased risk need early identification and medical guidance. In the past two decades, researchers have invested a lot of effort to develop clinical decision support systems for risk assessment of CVDs, but only some are included in clinical guidelines [2].

In this paper we tackle the issue of developing a patient risk stratification model, which classifies patients into levels for having a serious cardiac event. To improve the predictive performance we utilize ideas from the field of semi-supervised learning, which have shown that utilization of additional unlabeled data can improve accuracy of supervised learning models. Applying unlabeled data has advantages, such as not relying on expertise to label examples, which saves time, effort and reduces cost [12].

To perform knowledge transfer from unlabeled examples to supervised learning, we apply our implementation of the following four approaches: traditional semi-supervised learning as the baseline (SSL), active learning (AL), fuzzy learning (FL), and supervised clustering (SC). In the latter three approaches we pay special attention on how to perform the

instance knowledge transfer to properly select the right examples for the training data.

The paper is structured as follows. Related work is presented in Section 2. Our approaches are described in Section 3. The evaluation and results are given in Section 4. The last section concludes the paper and gives directions for future work.

2. RELATED WORK

The rapid development of predictive models for cardiac and cardiovascular disease diagnostics happened between years 2000 and 2013 [6]. In European guidelines, Systematic COronary Risk Evaluation (SCORE) estimates the ten-year risk for a fatal cardiovascular event, such as heart attack, stroke, aneurysm of the aorta, by stratifying patients into four risk groups: low, moderate, high, and very high. Different databases contain different risk level definitions. To facilitate an initial approach to the problem, we have chosen a simpler public Cleveland database with two levels of risk. The most commonly used machine learning (ML) algorithms for heart disease diagnosis are: Support Vector Machine (SVM), Naive Bayesian classifier (Naive Bayes), Artificial Neural Network (ANN) and Decision tree (DT). Parthiban et al. [10] used SVM with RBF kernel and Naive Bayesian classifier for diagnosis of heart disease in diabetic patients. The accuracy of their approach was 94.6% and 74%, respectively. Dangare et al. [3] propose using ANN with an extended feature set for heart diseases. They included information about obesity and smoking as the risk factors for coronary heart disease. Das et al. [4] used the ensemble approach of three ANN with a tangent sigmoid function, single hidden layer, and 14 neurons. The experimental results gained 89% classification accuracy for heart disease diagnosis.

Lately, knowledge transfer (transfer learning) has become popular in the field of machine learning [9, 7]. The reasons for transferring knowledge are often associated with a lack of learning data for the target problem or with the time it takes to learn a new model. Partially labeled training data have shown to improve performance in machine learning [8]. Such data are often also easier and cheaper to obtain. Knowledge transfer approaches are also found in different medical fields: pneumonia risk assessment with multi-task learning, lifelong inductive learning in the field of heart disease and sequential inductive a model for knowledge transfer in the field of coronary artery disease diagnosis [7]. A problem that can occur during the transfer of knowledge is the so-called *neg-*

ative transfer, which harms the learning success for the new domain [9].

3. UTILIZATION OF UNLABELED EXAMPLES

In this section we present four approaches for learning from partially labeled data using knowledge transfer. Each approach uses a learning algorithm to derive the knowledge from a small portion of labeled data. This knowledge is then used to classify unlabeled examples, which afterwards supplement the original training data set to increase the final prediction accuracy.

3.1 Semi-Supervised Learning

A well-known approach of SSL is called self-learning [12]. It first trains on labeled examples, then classifies the unlabeled examples and combines the latter with the initially labeled data. The extended dataset is used for further supervised learning. This method allows us to build on top of it and serves as a baseline in our experiments.

3.2 Active Learning

Active learning (AL) can similarly work with partially labeled data [11]. Its main goal is to find unlabeled examples that have the greatest potential to improve performance and present them to the teacher (oracle) who does the annotation process. The labeling of example is thus done iteratively rather than for all unlabeled dataset at once.

When selecting examples, we desire such that are labeled the most reliably. To estimate this reliability, we apply two metrics: (1) posterior class probability and (2) local modeling of prediction error with estimate CNK [1]. The posterior class probability for a given an example is provided by the learning algorithm. The reliability measure CNK estimates the reliability of the prediction by observing the local prediction error. In this work we adapt the original CNK estimate (designed for regression) for classification and compute it for each query example as:

$$CNK = \frac{\sum_i^k P_R(C_i)}{k}$$

where where C_i is an example from the local neighborhood $\{C_1, \dots, C_k\}$ of our query example, and $P_R(C_i)$ is the posterior probability that the neighbor is classified into R , which is the class into which our query example is also classified. To summarize, CNK measures the average posterior probability for the classification into query example’s class within its neighborhood. Such CNK estimate is defined on the interval $[0, 1]$, where 0 or 1 indicate unreliable or reliable classifications, respectively. In our experiments, we applied the size of the neighborhood $k = 5$, as used in the authors’ original work.

The algorithm stops when any of the following three stopping criteria is reached. The first criterion defines the maximum number of iterations (N), which can be useful for large data sets. The second criterion stops the algorithm when there are no examples with reliable classifications. The third criterion is fulfilled when all the unlabeled examples have been utilized.

3.3 Fuzzy Learning

Our Fuzzy learning (FL) approach labels examples with probabilities for belonging to all possible classes. We further use these probabilities to assign class probabilities to unlabeled examples as weighted class probabilities of the nearest neighbors using the locally weighted regression (LWR). We observe each class separately and assign a fuzzy class probability to each unlabeled example. A fuzzy class probability is derived from the class probabilities of local neighbors, which are weighted with the distance to the observed example and then summed up.

The weighted probabilities are afterwards calibrated to scale up to 1, to ensure probabilistic interpretation. We use measures of reliability, such as posterior class probability and local modeling of prediction error (CNK, as already described), to select examples which we include into the training set across multiple iterations. Finally, the learning algorithm is trained on the combined training set.

The algorithm stops either when a maximum number of iterations is reached, it runs out of data, or when labeling is not reliable enough to extend the training set.

3.4 Supervised Clustering

Supervised clustering (SC) differs from classical clustering methods by considering class values during the clustering process [13]. Our approach looks for representative examples in the available data. We assign each example a class of the closest representative. All examples are then used to train the learning algorithm.

To find the representatives, we apply the iterative SRIDHCR algorithm [5]. The algorithm first constructs a random set of representatives, which represents the current solution. In each iteration, a single non-representative is added and another single representative is removed, generating two new candidate sets of representatives. Next, the algorithm evaluates each generated candidate set X using the fitness function $q(X)$:

$$q(X) = \text{impurity}(X) + \beta * \begin{cases} \sqrt{\frac{|K| - c}{N}} & |K| > c \\ 0 & |K| \leq c \end{cases}$$

which minimizes cluster impurity and punishes large number of clusters. The punishment is controlled with the input parameter β and takes effect if the number of clusters $|K|$ is higher than the number of classes c . The candidate set, which improved the current best solution, is saved. In the next iteration, the procedure repeats itself until the algorithm cannot find a better set of representatives. The algorithm also utilizes a parameter S_{size} , which controls the number of candidate sets generated in each iteration. A higher value increases the probability of finding a better set of representatives because the algorithm performs more permutations.

Finally, we use the set of representatives from SRIDHCR to label the unlabeled data using the nearest neighbor approach. The final model is then trained on the combined data.

4. EVALUATION AND RESULTS

We evaluated our approaches on the Cleveland heart disease data from the UCI ML repository. The dataset has 297 patients of which 54% belong to low-risk (healthy) class and

46% to high-risk class. Two thirds of of patients are male with the average age of 54 years, and the remaining are female with the average age of about 56 years. We measured the performance using the Area Under a ROC Curve (AUC), which summarizes the overall performance of the model and reflects the discriminating ability to diagnose patients with and without the disease.

At the beginning of the evaluation process, we randomized the data. Next, the data was split using the 5-fold cross-validation into training and test sets. Since our experimental data set does not contain unlabeled examples, we split the training set into the labeled and unlabeled set (simulated, by hiding examples' classes). The ratio between the labeled and unlabeled set is controlled with an input parameter. In the experiments, we limited the AL and FL to 10 iterations and set their threshold to select examples with reliability at least 0.8. The SC generates 10 candidate sets for the representatives. The penalty (β) is set to 1.0 to prevent large numbers of representatives. The SRIDHCR algorithm performs restarts 30 times during the search process.

We used four different learning algorithms – Decision Tree (DT, using information gain and minimum number of 20 examples in leaves), K-Nearest Neighbors (KNN, with $k = 5$ and using Euclidean distance), Naive Bayesian classifier (NB) and Support Vector Machine (SVM, with linear kernel, regularization weight of 1 and termination criterion of 0.001). For each combination of approach and the learning algorithm we computed the *transductive* and *inductive* performance. The former reflects the quality of *transfer learning* by measuring labeling accuracy only of unlabeled examples prior their inclusion into the original learning data set. The inductive performance measures the final performance of the model that was built on the extended data set.

The transductive performance is shown in Table 1 and the inductive performance in Table 2. The first column displays the percentage of labeled examples that were used in the experiments. Approaches AL and FL are displayed twice (with the use of posterior probability and with the use of the CNK estimate). The results of the transductive analysis show that on the average FL with CNK selection method obtains the highest AUC. Using 20% initially labeled examples, the obtained AUC is equal to 0.85 ± 0.03 and increases to 0.90 ± 0.05 , with 80% of labeled examples. The AL approach shows low-averaged performance in combination with SVM, which in some cases learned to predict the opposite classes. The SC obtained the lowest transductive performance of 0.65 ± 0.14 on 20% and 0.75 ± 0.13 on 80% of labeled examples.

The inductive evaluation resembles the results of the transductive evaluation. The results of all approaches have decreased compared to transductive for about 0.05. FL obtained the best results followed by AL. The significant difference compared to the transductive results can be seen for SC. The predictive models obtain comparable performance, even though many misclassified examples were introduced. Using the 80% of initially labeled examples, the SC obtained AUC of 0.81 ± 0.05 , which is equal to the other approaches.

5. CONCLUSIONS

We experimented with four different approaches for including unlabeled examples into risk stratification. The obtained

results are comparable to the results in related works. The utilization of additional examples shows promising results, especially with the fuzzy learning approach that utilizes reliability estimate CNK.

In the future, we shall evaluate the methodology on databases with more complex risk levels. Secondly, we shall also analyze the performance of supervised learning algorithms and impact of their parameters. Thirdly, neural networks and deep learning are opening promising directions also in medical problems. Due to the limitation of resources, we did not include them in this work, but shall also include them in the work to follow.

6. REFERENCES

- [1] Z. Bosnić and I. Kononenko. Comparison of approaches for estimating reliability of individual regression predictions. *Data & Knowledge Engineering*, 67(3):504–516, 2008.
- [2] J. A. Damen, L. Hooft, E. Schuit, T. P. Debray, G. S. Collins, I. Tzoulaki, C. M. Lassale, G. C. Siontis, V. Chiochia, C. Roberts, et al. Prediction models for cardiovascular disease risk in the general population: systematic review. *bmj*, 353:i2416, 2016.
- [3] C. S. Dangare and S. S. Apte. Improved study of heart disease prediction system using data mining classification techniques. *International Journal of Computer Applications*, 47(10):44–48, 2012.
- [4] R. Das, I. Turkoglu, and A. Sengur. Effective diagnosis of heart disease through neural networks ensembles. *Expert systems with applications*, 36(4):7675–7680, 2009.
- [5] C. F. Eick, N. Zeidat, and Z. Zhao. Supervised clustering-algorithms and benefits. In *16th IEEE International Conference on Tools with Artificial Intelligence*, pages 774–776. IEEE, 2004.
- [6] M. Fatima and M. Pasha. Survey of machine learning algorithms for disease diagnostic. *Journal of Intelligent Learning Systems and Applications*, 9(01):1, 2017.
- [7] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, and G. Zhang. Transfer learning using computational intelligence: a survey. *Knowledge-Based Systems*, 80:14–23, 2015.
- [8] N. Nguyen and R. Caruana. Classification with partial labels. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 551–559. ACM, 2008.
- [9] S. J. Pan and Q. Yang. A survey on transfer learning. *10(22):1345–1359*, 2010.
- [10] G. Parthiban and S. Srivatsa. Applying machine learning methods in diagnosing heart disease for diabetic patients. *International Journal of Applied Information Systems (IJAIS)*, 3:2249–0868, 2012.
- [11] B. Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.
- [12] I. Triguero, S. García, and F. Herrera. Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study. *Knowledge and Information Systems*, 42(2):245–284, Feb 2015.
- [13] N. Zeidat, C. F. Eick, and Z. Zhao. *Supervised clustering: algorithms and applications*. University of Houston, 2005.

LABELS	METHOD	DT	KNN	NB	SVM	\bar{x}	p-value
20%	SSL	0.71 ± 0.05	0.80 ± 0.02	0.77 ± 0.06	0.79 ± 0.04	0.77 ± 0.04	-
	AL	0.75 ± 0.06	0.87 ± 0.03	0.79 ± 0.05	0.42 ± 0.42	0.70 ± 0.14	0.017
	AL (CNK)	<u>0.83 ± 0.04</u>	<u>0.82 ± 0.03</u>	<u>0.84 ± 0.03</u>	<u>0.68 ± 0.20</u>	<u>0.79 ± 0.07</u>	< 0.001
	FL	<u>0.81 ± 0.05</u>	0.81 ± 0.05	<u>0.81 ± 0.05</u>	0.81 ± 0.05	<u>0.81 ± 0.05</u>	< 0.001
	FL (CNK)	<u>0.85 ± 0.03</u>	<u>0.85 ± 0.03</u>	<u>0.85 ± 0.03</u>	<u>0.85 ± 0.03</u>	<u>0.85 ± 0.03</u>	< 0.001
	SC	0.64 ± 0.15	0.64 ± 0.13	0.67 ± 0.14	0.66 ± 0.13	0.65 ± 0.14	< 0.001
	\bar{x}	0.77 ± 0.06	0.80 ± 0.05	0.79 ± 0.06	0.70 ± 0.15	-	-
50%	SSL	0.75 ± 0.05	0.82 ± 0.02	0.79 ± 0.05	0.81 ± 0.02	0.79 ± 0.03	-
	AL	0.82 ± 0.06	0.88 ± 0.03	0.84 ± 0.05	0.66 ± 0.39	0.80 ± 0.13	< 0.001
	AL (CNK)	<u>0.87 ± 0.03</u>	<u>0.85 ± 0.03</u>	<u>0.87 ± 0.03</u>	0.74 ± 0.27	<u>0.83 ± 0.09</u>	< 0.001
	FL	<u>0.86 ± 0.03</u>	0.86 ± 0.03	<u>0.86 ± 0.03</u>	0.86 ± 0.03	<u>0.86 ± 0.03</u>	< 0.001
	FL (CNK)	<u>0.88 ± 0.02</u>	<u>0.88 ± 0.02</u>	<u>0.88 ± 0.02</u>	<u>0.88 ± 0.02</u>	<u>0.88 ± 0.02</u>	< 0.001
	SC	0.70 ± 0.13	0.72 ± 0.12	0.73 ± 0.12	0.72 ± 0.13	0.72 ± 0.13	< 0.001
	\bar{x}	0.81 ± 0.06	0.84 ± 0.04	0.83 ± 0.05	0.79 ± 0.14	-	-
80%	SSL	0.77 ± 0.06	0.81 ± 0.05	0.81 ± 0.07	0.82 ± 0.06	0.80 ± 0.06	-
	AL	0.82 ± 0.07	0.90 ± 0.04	0.86 ± 0.05	0.46 ± 0.46	0.76 ± 0.15	< 0.001
	AL (CNK)	<u>0.88 ± 0.05</u>	<u>0.87 ± 0.05</u>	<u>0.88 ± 0.05</u>	0.56 ± 0.43	<u>0.80 ± 0.15</u>	< 0.001
	FL	<u>0.87 ± 0.06</u>	0.87 ± 0.06	<u>0.87 ± 0.06</u>	0.87 ± 0.06	<u>0.87 ± 0.06</u>	< 0.001
	FL (CNK)	<u>0.90 ± 0.05</u>	<u>0.90 ± 0.05</u>	<u>0.90 ± 0.05</u>	<u>0.90 ± 0.05</u>	<u>0.90 ± 0.05</u>	< 0.001
	SC	0.75 ± 0.13	0.76 ± 0.11	0.72 ± 0.13	0.75 ± 0.16	0.75 ± 0.13	0.003
	\bar{x}	0.83 ± 0.07	0.85 ± 0.06	0.84 ± 0.07	0.73 ± 0.20	-	-

Table 1: **Transductive** AUC performance for different percentages of labeled examples, labeling approaches and four classifiers. Statistically significant differences to the baseline (SSL) approach are underlined.

LABELS	METHOD	DT	KNN	NB	SVM	\bar{x}	p-value
20%	BASE	0.72 ± 0.07	0.80 ± 0.05	0.78 ± 0.06	0.80 ± 0.06	0.77 ± 0.06	-
	SSL	0.72 ± 0.07	0.80 ± 0.05	0.79 ± 0.06	0.80 ± 0.06	0.77 ± 0.06	0.938
	AL	0.72 ± 0.08	0.83 ± 0.05	<u>0.76 ± 0.06</u>	<u>0.66 ± 0.16</u>	0.74 ± 0.09	0.032
	AL (CNK)	0.72 ± 0.08	0.81 ± 0.05	0.80 ± 0.05	0.80 ± 0.07	0.78 ± 0.06	0.066
	FL	<u>0.76 ± 0.07</u>	0.79 ± 0.05	0.77 ± 0.05	0.78 ± 0.06	0.78 ± 0.06	0.881
	FL (CNK)	<u>0.78 ± 0.05</u>	0.80 ± 0.04	0.79 ± 0.05	0.80 ± 0.06	<u>0.79 ± 0.05</u>	0.016
	SC	<u>0.65 ± 0.15</u>	0.67 ± 0.16	<u>0.77 ± 0.08</u>	<u>0.68 ± 0.15</u>	<u>0.69 ± 0.14</u>	< 0.001
	\bar{x}	0.72 ± 0.08	0.78 ± 0.06	0.78 ± 0.06	0.76 ± 0.09	-	-
50%	BASE	0.73 ± 0.07	0.80 ± 0.03	0.78 ± 0.06	0.81 ± 0.05	0.78 ± 0.05	-
	SSL	0.73 ± 0.07	0.81 ± 0.04	0.79 ± 0.05	<u>0.80 ± 0.06</u>	0.78 ± 0.06	0.394
	AL	0.74 ± 0.08	0.81 ± 0.04	0.79 ± 0.05	0.80 ± 0.05	0.79 ± 0.06	0.993
	AL (CNK)	<u>0.76 ± 0.07</u>	0.81 ± 0.05	0.80 ± 0.05	0.81 ± 0.05	0.79 ± 0.05	0.071
	FL	<u>0.79 ± 0.05</u>	0.77 ± 0.05	0.81 ± 0.05	0.82 ± 0.05	0.80 ± 0.05	0.087
	FL (CNK)	<u>0.82 ± 0.04</u>	<u>0.78 ± 0.05</u>	<u>0.81 ± 0.05</u>	0.81 ± 0.05	<u>0.81 ± 0.05</u>	0.002
	SC	0.71 ± 0.10	0.77 ± 0.08	0.80 ± 0.05	<u>0.75 ± 0.12</u>	0.76 ± 0.09	0.102
	\bar{x}	0.75 ± 0.07	0.79 ± 0.05	0.80 ± 0.05	0.80 ± 0.06	-	-
80%	BASE	0.77 ± 0.05	0.83 ± 0.04	0.81 ± 0.05	0.83 ± 0.06	0.81 ± 0.05	-
	SSL	0.77 ± 0.05	0.83 ± 0.04	0.82 ± 0.05	0.82 ± 0.05	0.81 ± 0.05	0.851
	AL	0.78 ± 0.04	0.83 ± 0.04	0.82 ± 0.05	0.82 ± 0.05	0.81 ± 0.05	0.334
	AL (CNK)	0.77 ± 0.04	0.83 ± 0.04	0.82 ± 0.05	0.83 ± 0.05	0.81 ± 0.05	0.680
	FL	<u>0.81 ± 0.05</u>	0.78 ± 0.07	0.83 ± 0.05	0.83 ± 0.06	0.81 ± 0.06	0.651
	FL (CNK)	<u>0.80 ± 0.03</u>	<u>0.78 ± 0.05</u>	<u>0.83 ± 0.05</u>	0.83 ± 0.05	0.81 ± 0.05	0.899
	SC	0.78 ± 0.05	0.83 ± 0.06	0.82 ± 0.06	0.82 ± 0.05	0.81 ± 0.05	0.818
\bar{x}	0.78 ± 0.05	0.82 ± 0.05	0.82 ± 0.05	0.83 ± 0.05	-	-	
100%	BASE	0.79 ± 0.04	0.82 ± 0.04	0.82 ± 0.05	0.83 ± 0.03	0.82 ± 0.04	-

Table 2: **Inductive** AUC performance for different percentages of labeled examples, labeling approaches and four classifiers. Statistically significant differences to the baseline (BASE - a model trained on initially labeled data) approach are underlined.

Fitness-Based Student Clustering Combining Clustering Algorithms and Dimensionality Reduction

Erik Dovgan
Department of Intelligent Systems
Jožef Stefan Institute
Jamova cesta 39
SI-1000 Ljubljana, Slovenia
erik.dovgan@ijs.si

Mitja Luštrek
Department of Intelligent Systems
Jožef Stefan Institute
Jamova cesta 39
SI-1000 Ljubljana, Slovenia
mitja.lustrek@ijs.si

ABSTRACT

Health status and well-being of persons are significantly influenced by their physical fitness since, for example, low fitness is related to several health risks. Identification of unfit persons enables us to personalize advice or recommendations to improve their fitness. However, this identification is not straightforward or standardized. For this purpose, we propose a combination of dimensionality reduction methods and clustering algorithms on data from a test battery used in schools. Such an approach enables teachers, parents and policy makers to identify clusters of fit and unfit students, and better target actions for improving student fitness.

Keywords

Clustering algorithms, Dimensionality reduction, Student clustering

1. INTRODUCTION

Physical fitness has significant impact on health and well-being, since several health risk factors are related to low fitness (see examples in [4] and [2]). In order to reduce the risks, the physical fitness should be improved. This can be achieved, for example, in schools, where teachers, parents and policy makers can create and provide advice and recommendations. Although general advice and recommendations are possible, they are more efficient, when they are personalized and better-targeted. However, to achieve this, we firstly need to identify unfit students, which is not trivial.

Identification of fit and unfit students is not straightforward or standardized. There are some metrics, such as body mass index (BMI) [3] and the related Overweight and Obese Adolescents (OOA) categories [1], which enable identifying underweight, normal weight, overweight, and obese persons. However, these metrics are not directly related to the physical fitness and consequently cannot be effectively used to identify unfit students. To cluster students into fit and unfit, we propose to apply dimensionality reduction methods and clustering algorithms on data from widely used test battery. The identification of unfit students will enable decision makers to personalize actions for improving students' fitness.

The paper is further organized as follows. The procedure for identification of fit and unfit students is presented in Section 2. Section 3 describes the experiments in terms of the used dataset and the obtained results. Finally, Section 4 concludes the paper with ideas for future work.

2. IDENTIFICATION OF FIT AND UNFIT STUDENTS

The goal of the developed procedure is the identification of fit and unfit students. Since there are different risks between underweight, normal weight and overweight students, we focus only on one of these categories, namely overweight students. Note that this category also includes obese persons.

The developed procedure takes into account a set of physical fitness measurements, known as SLOfit test battery¹, which are performed yearly in Slovenian schools. The set of measurements is shown in Table 1. These attributes are given as raw data or as percentiles, where the attribute's percentile, i.e., the attribute's quantile, is the rank of the student based on this attribute within the set of students of the same sex and age. In addition to the measurement data, the procedure can also take into account a set of additional attributes that are shown in Table 2. Note that Fitness index (FI) is the quantile rank of the sum of the quantile ranks of the fitness measurements.

Table 1: Measurements of the test battery.

Measurement	Short name
Thickness triceps skinfold	TTSF
Reaction time during arm plate tapping	TAPT
Distance during standing broad jump	DSBJ
Time pass polygon backwards on all fours	TOCB
Number sit-ups in 60 s	NSU
Distance fingertips-toes, bending forward	DSR
Time bent arm position, hanging from bar	TBAH
Time run 60 m	T60m
Time run 600 m	T600m

The proposed procedure searches for fit and unfit students as follows. The input data consist of the measurement attributes and (a subset of) additional attributes. These data are clustered into groups of students. However, the true clusters are not given thus the quality of clustering cannot be easily assessed. As a solution, we apply dimensionality reduction to obtain two-dimensional representation of data, which is then visually assessed in terms of meaningfulness of the obtained clusters. The meaningfulness of the clusters is assessed based on two clusters' properties:

¹<http://en.slofit.org/measurements/test-battery>

Table 2: Additional attributes.

Attribute	Short name
Sex	SEX
Height (raw or percentile)	H
Weight (raw or percentile)	W
Grade	GRD
Age	AGE
Fitness index	FI
OOA categories	OOA
Body Mass Index	BMI

1. We aim at obtaining at least two clusters that are separable in the reduced-dimensional space.
2. The obtained clusters should not be correlated to discrete attributes, i.e., SEX, GRD, AGE, and OOA. Such correlation is not wanted due to the fact that the easiest way to cluster or reduce dimensions is to focus on attributes that are already separable, e.g., discrete attributes. However, such a clustering/dimensionality reduction is meaningless for decision making, e.g., it makes no sense to find clusters of males and females since these clusters are already known.

In our procedure, we apply the following clustering algorithms: KMeans, Affinity Propagation, Mean Shift, and Birch. In addition, we use the following dimensionality reduction methods: Factor analysis (FA), Principal component analysis (PCA), Singular value decomposition (SVD), Independent component analysis (ICA), Isometric mapping (ISOMAP), and Uniform manifold approximation and projection (UMAP). The developed procedure works as follows. For each combination of clustering algorithms and dimensionality reduction methods we apply the following steps.

Step 1: A subset of data is randomly selected to make clustering and dimensionality reduction feasible (due to the fact that some methods are computationally intensive).

Step 2: This subset is clustered and the model for data clustering is obtained.

Step 3: Dimensionality reduction method is applied on the subset and the model for dimensionality reduction is obtained.

Step 4: The entire dataset is clustered with the clustering model.

Step 5: The model for dimensionality reduction is applied on the entire dataset.

Step 6: The entire dataset is presented in the reduced-dimensional space. Clusters are marked with different colors.

Step 7: This representation is visually assessed in terms of meaningfulness of the obtained clusters.

3. EXPERIMENTS AND RESULTS

The proposed procedure was evaluated on two relevant sets of students from the SLOfit dataset: a) High school students (ages 15–19), and b) Elementary school students (up to age of 11). In addition, only data from the most recent year was used, i.e., 2018. The attributes are shown in Tables 1–2. Note that the data of high school students included GRD,

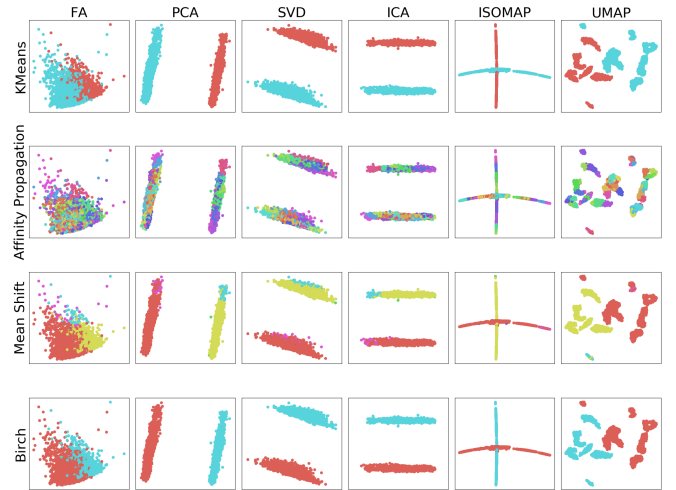


Figure 1: Dimensionality reduction and clustering on nonpercentile data from high school students. Reduced dimensions are presented with position in the new space, while clusters are shown with different colors.

but not AGE, while data of elementary school students included AGE, but not GRD. The following sections present the analysis of the data, which was performed in two steps.

3.1 Initial Analysis

The initial analysis was performed on all the attributes and with all the clustering algorithms and dimensionality reduction methods. The results are presented in Figure 1. This figure shows that several dimensionality reduction methods found two (separable) clusters, i.e., PCA, SVD, and ICA. In addition, KMeans and Birch identified those two clusters. However, the highest correlation with these clusters are obtained by SEX (see Figure 2). In addition, Figure 3 confirms that the two clusters obtained with reduced dimensions represent two sexes. Such clusters are obvious and thus not interesting for the decision makers.

Additional tests were performed on subsets of attributes and the results showed that only some subsets produced interesting clusters. Therefore, we decided to systematically assess various subsets of attributes. Since the measurements (see Table 1) are (probably) the most suitable for determining the physical fitness, we evaluated only the subsets of additional attributes (see Table 2), while measurements were always considered.

The results also indicated that some clustering algorithms and dimensionality reduction methods were redundant or uninformative. For example, Mean Shift and Affinity Propagation found more clusters than needed, while KMeans and Birch discovered the best (but the same) clusters (see Figure 1). Therefore, KMeans and Birch are redundant and we prefer KMeans among them due to its simplicity. In addition, dimensionality reduction methods can be divided into two groups: a) components/factor based (FA, PCA, SVD, and ICA), and b) projection based (ISOMAP and UMAP). The former, for example, aim at maximizing the variance

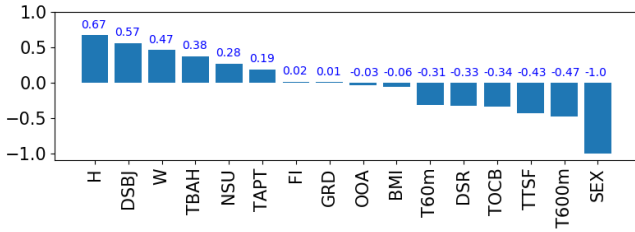


Figure 2: Correlation of clusters found by KMeans with the attributes on nonpercentile data from high school students.

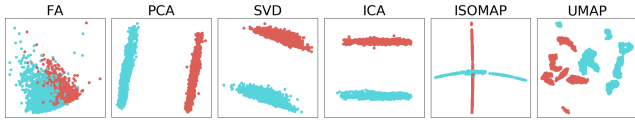


Figure 3: Relation between data in reduced dimensions and the SEX attribute on nonpercentile data from high school students, where SEX is shown with different colors.

in the dataset (PCA), while the latter try to maintain the distances between the data. Therefore, one representative of each group should be used, i.e., PCA as the most widely used from the first set, and UMAP (that is faster than ISOMAP) from the second set.

3.2 Systematic Analysis

Based on the results from the initial analysis, we decided to systematically evaluate all subsets of additional attributes (see Table 2) using only KMeans for clustering, and PCA and UMAP for reducing dimensions. Note that measurements' data (see Table 1) were always taken into account. In this way, the number of tested subsets of attributes was $2^7 = 128$ (due to 7 additional attributes). Each subset of attributes was evaluated four times: 1) percentile attributes of elementary school data, 2) nonpercentile attributes of elementary school data, 3) percentile attributes of high school data, and 4) nonpercentile attributes of high school data. This resulted in $4 \times 128 = 512$ tested subsets.

To additionally simplify clustering and find meaningful clusters, we grouped subsets of attributes with respect to their results and cluster the data with respect to clusters obtained with UMAP. More precisely, the procedure was as follows:

- Step 1:** Reducing dimensionality with UMAP.
- Step 2:** Clustering students for all subsets of attributes. Each subset represents one instance. Many instances cluster students similarly, therefore instances should be grouped in order to find only representative instances, i.e., one instance for each group of instances (Step 3).
- Step 3:** Grouping similar instances with respect to clusters of discrete attributes, i.e., OOA, SEX, and GRD/AGE, based on visual inspection. For example, grouping together instances with two clusters which represent two SEX-es, or instances with four clusters which represent four GRD-es.
- Step 4:** Clustering students for each group found in Step 3. The input for this clustering are the cluster ids of instances

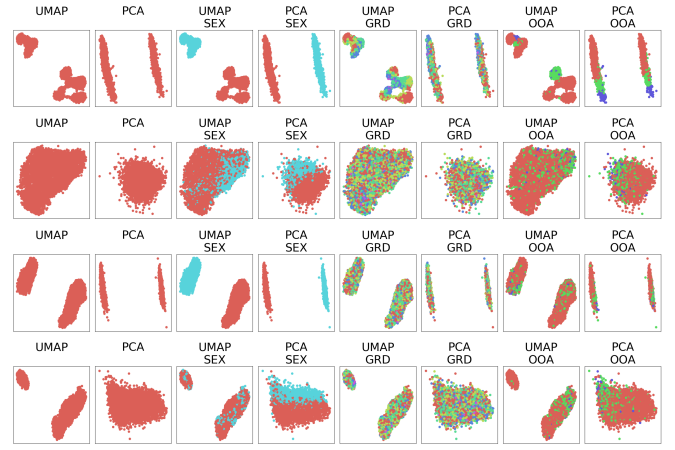


Figure 4: Examples of clusters found by PCA and UMAP on nonpercentile data from high school students. Columns 3–8 show the data with respect to discrete attributes, where attribute values are represented with different colors. Rows show the clusters with respect to four set of attributes that were used by dimensionality reduction methods. First row: all attributes, clusters represent SEX; Second row: measurements only, no clusters found; Third row: measurements and SEX, clusters represent SEX; Fourth row: measurements and FI, no connection between clusters and discrete attributes. According to definition of meaningful clusters given in Section 2, only the fourth row shows good clusters.

within the group, where these cluster ids were found in Step 2. The instances within the group might not produce similar clusters of students, therefore, such non-homogenous groups should be split (Step 5).

Step 5: Visual inspection of the obtained clusters for each group. If the found clusters are not consistent among the instances, the group is split and Step 3 is repeated. For example, if the group contains instances with two clusters and only half of these instances are correctly clustered into two clusters (i.e., consistent with clustering for the whole group), the correctly clustered instances represent a good subgroup and Step 6 should be applied. On the other hand, the incorrectly clustered instances are not a good subgroup and Step 3 has to be repeated.

Step 6: Finding representative instances of each group found in Step 5. For example, if attributes (e.g., SEX, GRD, W, H) of first instance are subset of the attributes of the second instance, the first instance (with a lower number of attributes) is more representative and should be used for further analysis.

This procedure enabled us to reduce the number of instances, i.e., subsets of attributes, that need to be analyzed, since it found the most representative subsets of attributes only.

The representative clusters are described in Table 3 in terms of additional attributes that were used (in addition to measurements), attribute type (raw or percentiles), the number of obtained clusters and the attribute with the highest cor-

Table 3: Properties of representative subsets of attributes obtained by combining dimensionality reduction methods and clustering algorithms.

Students	Additional attributes	Attribute type	Number of clusters	Attribute with highest correlation
high school	OOA	raw	2	OOA
high school	GRD, H, BMI	raw	2	GRD
high school	SEX	raw	2	SEX
high school	FI	raw	2	FI
high school	OOA, BMI	percentile	2	OOA
high school	SEX	percentile	2	SEX
high school	FI, GRD, W	percentile	2	FI
elementary school	AGE, H, W, FI, BMI, OOA	raw	2	OOA
elementary school	AGE, H, W, FI, BMI	raw	2	AGE
elementary school	SEX, AGE, W, FI, OOA	raw	3	SEX
elementary school	AGE, W, BMI, OOA	raw	3	OOA
elementary school	SEX, H, FI, BMI, OOA	raw	4	OOA
elementary school	AGE, W, FI	raw	5	AGE
elementary school	SEX, AGE, H, BMI, OOA	raw	6	SEX
elementary school	SEX, AGE, H, W, BMI, OOA	raw	7	SEX
elementary school	SEX, AGE, W, OOA	raw	8	SEX
elementary school	SEX, AGE, BMI, OOA	raw	10	SEX
elementary school	SEX, BMI, OOA	percentile	2	SEX

relation with clusters. Examples of obtained clusters are shown in Figure 4.

The results in Table 3 show that the majority of the obtained clusters is correlated with discrete attributes such as OOA, GRD/AGE, and SEX. The only exceptions are the two subsets of attributes whose clusters are correlated with FI. These subsets are also the most interesting ones, since the goal was to cluster the students based on their fitness into fit and unfit groups. These interesting clusters can be also seen in Figure 4 in fourth row. This figure confirms that the obtained clusters are not correlated with discrete attributes, i.e., SEX, GRD, and OOA.

Although we were able to find interesting clusters, i.e., those that are not correlated with discrete attributes, these clusters were obtained by using only specific subsets of attributes and only for high school data (see Table 3). In addition, no interesting clusters were obtained from elementary school data. It should be also noted that out of 512 possible attribute subsets, only two subsets were interesting, i.e., produced at least two clusters that were not correlated with discrete attributes.

4. CONCLUSION

This paper presented an approach for identification of fit and unfit students. This approach analyzes data from test battery used in schools by combining dimensionality reduction methods and clustering algorithms. By visually inspecting the results of data analysis, it enables us to find combinations of attributes that produce meaningful clusters of fit and unfit students. The identification of unfit students supports teachers, parents and policy makers in better targeting actions for improving fitness of those students.

In our future work, we will aim at developing a method for automatic assessment of obtained clusters, which will be

used instead of visual assessment. To this end, an analytical approach for assessing the quality of clusters will be developed. This approach will also aim at determining to which extent the obtained clusters represent fit and unfit students. This task is especially challenging since we do not have the true values.

5. ACKNOWLEDGMENTS

This work is part of a project that has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement no. 727560. We would also like to acknowledge the help of Bojan Leskošek, Gregor Jurak, Gregor Starc, and Maroje Sorić, from the Faculty of Sport, University of Ljubljana, Slovenia, who provided the SLOfit dataset and helped defining the problem.

6. REFERENCES

- [1] F. Bacha, R. Saad, N. Gungor, J. Janosky, and S. A. Arslanian. Obesity, regional fat distribution, and syndrome X in obese black versus white adolescents: Race differential in diabetogenic and atherogenic risk factors. *The Journal of Clinical Endocrinology and Metabolism*, 88:2534–2540, 2003.
- [2] S. W. Farrell, C. E. Finley, N. B. Radford, and W. L. Haskell. Cardiorespiratory fitness, body mass index, and heart failure mortality in men. *Circulation: Heart Failure*, 6(5):898–905, 2013.
- [3] M. Kallioinen and S. I. Granheim. Overweight and obesity in the western pacific region. Technical report, World Health Organization, 2017.
- [4] J. R. Ortlepp, J. Metrikat, M. Albrecht, P. Maya-Pelzer, H. Pongratz, and R. Hoffmann. Relation of body mass index, physical fitness, and the cardiovascular risk profile in 3127 young normal weight men with an apparently optimal lifestyle. *International Journal of Obesity*, 27:979–982, 2003.

Counting Bites with a Smart Watch

Mitja Luštrek^{1,2}, Benjamin Fele¹, Nina Reščič^{1,2}, Vito Janko^{1,2}

¹ Jožef Stefan Institute, Department of Intelligent Systems

² Jožef Stefan Postgraduate School

Jamova cesta 39, 1000 Ljubljana

Slovenia

{mitja.lustrek, nina.rescic, vito.janko}@ijs.si, benjamin.fele@gmail.com

ABSTRACT

The work described in this paper is a part of the WellCo project, which is developing a virtual coach for healthy lifestyle. An important aspect of a healthy lifestyle is nutrition, and knowing as much as possible about the users' current nutrition can contribute to better coaching. We therefore set out to count the number of times the users take food to their mouths (bites) using smartwatch sensors. This enables identifying the meals as well as estimating the caloric intake and the speed of eating. We compare three approaches: two that rely on classical machine learning and hidden Markov models, and one that uses deep learning. The F-scores of the approaches range from 0.63 to 0.91, and the percentages of miscounted bites from 6.9 % to 10.7 %, with a different approach scoring best on each metric.

Keywords

Nutrition, monitoring, bite counting, wearables, smart watch, machine learning, hidden Markov models, deep neural networks

1. INTRODUCTION

The WellCo project is developing a virtual coach for seniors, which will provide advice on healthy lifestyle and wellbeing. To provide quality coaching and maximise the chances of achieving behaviour change, the advice should be fully personalised – not only adapted to the user's needs and wishes, but also to their current situation. To do so, the WellCo system uses smartphone and smart-watch sensors to monitor the users. One of the areas of coaching and therefore monitoring is nutrition. We want to know both what the users eat, as well as when and how they do it. The first part is addressed by questionnaires described elsewhere [1], while this paper deals with detecting eating and counting the number of times food is taken to the mouth (bites).

To count bites, the accelerometer and gyroscope in the smartwatch are used. These two sensors detect movement of the hand when the user is eating, and with the help of machine learning, these can be translated into individual bites. Section 2 briefly presents some related work on sensor-based nutrition monitoring, both using inertial sensors in wearables, as well as other approaches. In Section 3, we present the public dataset that was used to train and evaluate our methods. In Section 4, we describe three approaches to bite recognition and counting, starting with the simplest and ending with one using two (modestly) deep neural networks. Section 5 presents the experimental evaluation of the methods. Section 6 concludes the paper with a discussion of the integration of the described methods in the WellCo system, as well as some directions for future work.

2. RELATED WORK

The traditional tools for nutrition monitoring are questionnaires. However, these often prove inaccurate, especially regarding the quantity of food consumed – in one case it was underreported by up to 30 % for normal-weight subjects and 50 % for obese adults and children [2]. Therefore automated monitoring solutions are becoming increasingly important. By analysing photos of meals, one can determine the type and amount of food [3]. Using wearable sensors, it is possible to recognise the time, quantity and to some degree the type of food consumed in each bite. With development of smart watches and other (watch-like) wristbands, gesture recognition has been explored for this purpose [4][5]. With such devices, it is possible to recognise eating gestures, count bites and estimate the caloric intake. On-ear microphone or throat microphone can be used to detect chewing sounds [6][7], and swallows can be counted using a neck-worn sensor [8]. Out of these approaches, those relying on wrist-worn devices are the least intrusive and were thus selected for the WellCo system.

3. DATASET

We used the publicly available Food Intake Cycle (FIC) dataset (<https://mug.ee.auth.gr/intake-cycle-detection/>) in the research described in this paper. It contains triaxial signals from accelerometers and gyroscopes in wrist devices with the sampling frequency of 100 Hz. 21 meal sessions by 12 unique subjects were recorded in the restaurant of the university using two commercial devices: Microsoft Band 2 for 10 out of the 21 meals, and Sony Smartwatch 2 for the remaining meals (both were worn on the dominant hand). In addition, the start and end moments of each food intake cycle (bite) as well as of each micromovement was labelled throughout the dataset.

4. BITE RECOGNITION METHODS

The most straightforward approach to bite recognition is to adopt the method usually used for activity recognition: split the stream of sensor data into windows and recognise the activity in each window using a machine-learning model. These activities – when they are a part of the bite cycle – are termed micromovements in this paper. This method on its own is not sufficiently accurate, so in Section 4.1, we describe an extension that applies smoothing and other postprocessing. In Section 4.2, we describe the most commonly used approach for bite recognition, which uses two HMMs – one for bites and one for non-bites [9]. Micromovement sequences are fed into both and classified based on which HMM they fit better. In Section 4.3, we describe a conceptually similar approach that replaces the classical machine-learning model for micromovement recognition with one neural network and the HMMs with another [10].

4.1 Micromovement Recognition with Smoothing

4.1.1 Classical Micromovement Recognition

The first step of this approach was to recognise micromovements related to the bite cycle – No movement, Pick, Upwards, Mouth, Downwards and Other. The Other label was used for non-eating activities, such as gesticulating. We used a 0.2-s sliding window (0.1 s overlapping) to compute features. Time-domain features that proved themselves in our previous work [11][12] were used. These features were designed for accelerometer data, and most of them were calculated only on the acceleration (and derived) data streams. However, the features that were also meaningful for gyroscope data were calculated from those data streams as well. After the features were computed, a feature selection using the methodology from our previous work was performed to filter out the redundant and uninformative ones.

In the above-mentioned previous work, features were calculated on acceleration data filtered with low-pass and band-pass filters. In the present work, we also filtered accelerometer data with a low-pass filter, however, we used “relative acceleration” instead of the band-pass filter. This was proposed by the authors of the FIC dataset. We computed relative acceleration by subtracting the first element of each window from all values in the window of length n .

$$a_{rel}(i) = a(i) - a(1); i = 1, 2, \dots, n$$

The random forest algorithm was used to build the micromovement recognition model. We built two versions – the first using all six micromovements as possible class values, and the other using all the micromovements except Other. We opted for the latter in further steps of our approach since the recognition of Other proved very difficult and made the model highly inaccurate.

4.1.2 Smoothing and Other Postprocessing

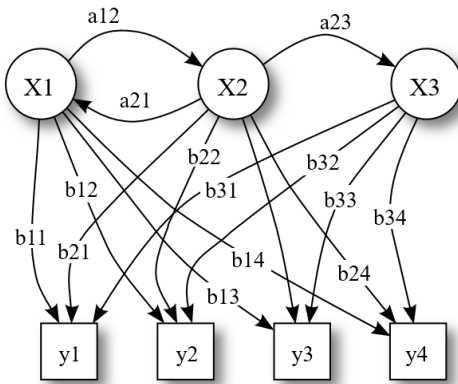


Figure 1: Example hidden Markov model

(licensed under Creative Commons Attribution 3.0 Unported license by Tdunning).

The second step of our approach was to smooth the recognised micromovements with a hidden Markov model (HMM). An HMM is defined by the following:

- Hidden states, which are labelled as X1 ... X3 in the example in Figure 1, and correspond to true micromovements in our approach. They are called

hidden because one cannot observe them directly – like true micromovements are hidden, otherwise our task would be trivial.

- Possible observations or emissions, which are labelled as y1 ... y4 in Figure 1. These correspond to the micromovements as recognised by the first step of our approach.
- Emission probabilities – each emission j has a probability of occurring in each hidden state i , which is labelled as b_{ij} in Figure 1. These probabilities correspond to the probabilities of recognising a micromovement i as micromovement j by the first step of our approach.
- Transition probabilities – these are the probabilities of transitioning between each pair of hidden states (micromovement in our case) i and j , which are labelled as a_{ij} in Figure 1.

We built an HMM to describe the training portion of the FIC dataset, with the emission probabilities set based on the results of the micromovement recognition, and the transition probabilities extracted from the dataset. On the test portion of the dataset, we used the Viterbi algorithm to compute the most probable sequence of hidden states corresponding to the observed emissions. This means that we computed the most probable sequence of true micromovements based on the recognised micromovements, or, in other words, that we smoothed the recognised micromovements.

The smoothed micromovements were still not all correct, particularly where the true micromovement was Other, since our micromovement-recognition model was not trained to recognise that. We therefore trained a dedicated model to recognise the Other micromovement. It worked on the outputs of the micromovement-recognition model. One instance for this model was a continuous segment in which the same micromovement was recognised by the micromovement-recognition model. The features were the probability of each class output by micromovement-recognition model averaged over the segment, the standard deviations of these probabilities, and the length of the segment. This Other-recognition model was tuned so that it had precision above 90 % (while the recall was only 28.1 %) – we wanted to correct only the micromovements for which we were very confident they are Other, since the final step of the approach was capable of dealing with many of the remaining mistakes.

The final step looked at each quartet of consecutive segments, and penalised them based on how much they deviated from the ideal bite quartet of micromovement segments Pick, Upwards, Mouth and Downwards. Each segment s in the quartet was penalised if its length was atypical:

$$penalty(s) = - \frac{|\text{length}(s) - \overline{\text{length}}|}{\delta_{\text{length}}}$$

Penalty of -2 was added to the quartet if one of the expected segments was missing or if an incorrect segment was inserted. More than one mistake of this type was not tolerated. In the end, each quartet with the penalty above the experimentally set threshold of -5.4 was considered a bite. An example of true and smoothed micromovements, and penalty, is shown in Figure 2.

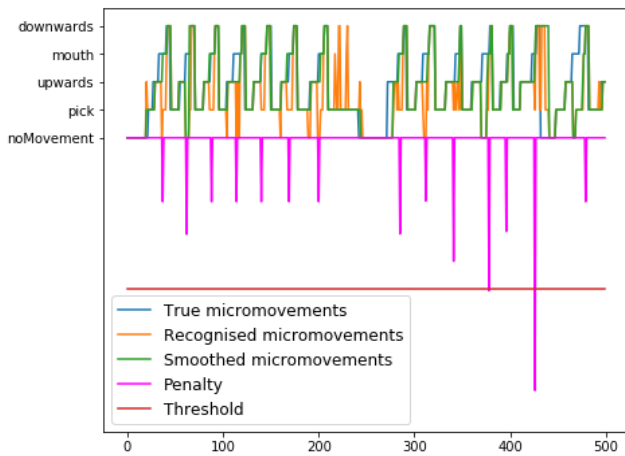


Figure 2: Bite recognition with micromovement recognition and smoothing.

4.2 Bite vs. Non-Bite HMM

The first step of this approach consisted of micromovement recognition as described in Section 4.1.1. The sequence of micromovements served as the input to the second step, which was implemented with two HMMs.

To build the HMMs for bite recognition, we randomly selected 1,000 bite and 1,000 non-bite instances from the FIC dataset. Each instance was 4 s long, which was the average length of a bite in the dataset. For an instance to be considered a bite, it needed to have an 80 % overlap with the ideal bite quartet of micromovement segments Pick, Upwards, Mouth and Downwards. For an instance to be considered a non-bite, it needed to have less than 60 % overlap with any such complete bite. The bite and non-bite datasets were then used to train (adjust parameters of) the two HMMs using the Baum-Welch algorithm. The number of hidden states in the models was experimentally set to 10.

After the models were built, we could pass over an input sequence of micromovements with a 4 s sliding window. We used the Forward-backward algorithm to estimate the probability that the content of the window was generated by the bite and non-bite model. We then subtracted the score returned for the non-bite model from the score returned for the bite model (the scores expressed log probabilities). The difference was proportional to the probability that the window contained a bite.

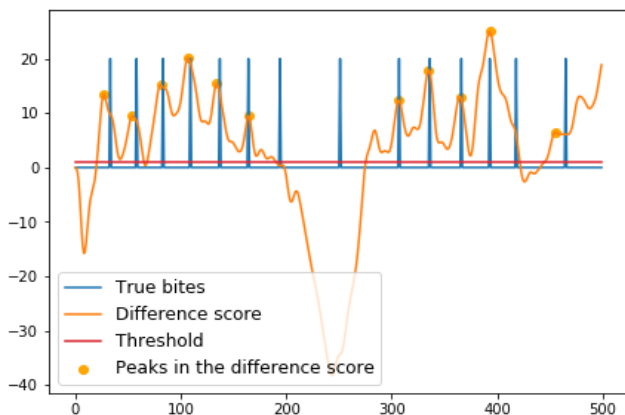


Figure 3: Bite recognition with bite vs. non-bite HMM.

The difference score was unfortunately not adequate to detect bites directly. Therefore, we first applied a Butterworth low-pass filter in a longer window to smooth it (5th order with a cut-off frequency of 1 Hz). Afterwards, we detected peaks in the score – each peak corresponds to one bite. For a peak to be detected, the score had to be larger than its neighbours, it had to be higher than an experimentally set threshold of 1, and it had to be at least 2.5 s from the previous peak. An example of true bites, the difference score and recognised bites is shown in Figure 3.

4.3 Deep Neural Network

4.3.1 CNN Micromovement Recognition

This step corresponds to the micromovement recognition from Section 4.1.1, except that a convolutional neural network was used instead of the random forest algorithm. A 0.2 s sliding window with a step of 0.1 s was used again. The input data were transformed using a median filter and a high-pass filter with the cut-off frequency of 1 Hz. The data were then normalized so that each data stream had the mean of 0 and standard deviation of 1.

The neural network consisted of two convolutional layers, each of them followed by a max pooling layer. The first convolutional layer used 64 filters, while the second used 128 filters, with both having the filter size set to 6. They were followed by a dropout and a fully-connected layer, after which the probability distribution of the five micromovements was retrieved with a softmax activation function. Categorical cross entropy was used as the loss function when training the model.

4.3.2 LSTM Bite Recognition

This step corresponds to the bite recognition from Section 4.2, except that a long short-term memory (LSTM) neural network was used instead of HMMs. A 3.6 s sliding window was used, which was the median length of a bite in the dataset, with a step of 0.1 s. The sequence of micromovement probability distributions from the first step of the approach was fed into two LSTM layers with 64 units each, again followed by a dropout and a fully-connected layer. The network output was a value gated using a sigmoid activation function. Binary cross entropy was used as the loss function when training the model.

To correctly recognise individual bites, we applied an experimentally set threshold of 0.87 to the output from the LSTM network. For each set of probabilities above the threshold, we found the maximum value, which denotes the bite moment. Then we disregarded all bites detected less than 2 s after the previous one. An example of the network’s output with the true and recognised bites is shown in Figure 4.

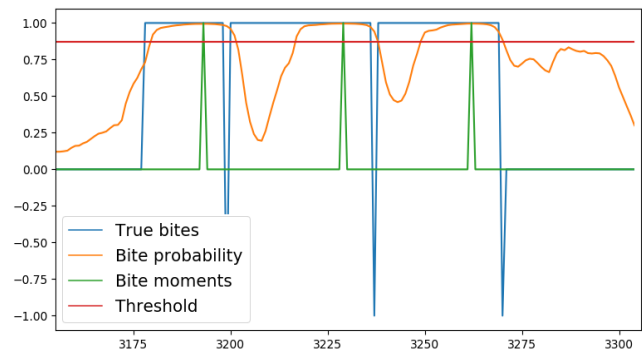


Figure 4: Bite counting with deep neural network.

5. EXPERIMENTAL EVALUATION

To evaluate the smartwatch-based nutrition monitoring, we again used the FIC dataset described in Section 3. The evaluation of the micromovement recognition used the was straightforward, using the leave-one-meal-out approach. This means that the feature selection and training of the model were performed on the data of all meals but one, and tested on the data of the remaining meal. The procedure was repeated for all the meals and the results averaged.

The evaluation of bite recognition was somewhat more involved. It was first evaluated in terms of precision and recall: precision is the fraction of instances recognised as bites that were in fact bites, while recall is the fraction of bite instances that were recognised as such. The first recognised bite inside each true bite interval was considered a true positive, and any other recognised bites inside that interval were considered false positives. Any true bite interval without recognised bites was considered a false negative. Bite recognition or bite counting was also evaluated in terms of the percentage by which it miscounted the number of bites in a meal. The three approaches described in Section 4 were compared to a baseline approach that considered every segment with the Mouth micromovement a bite. The results are shown in Table 1.

	Microm. only	Microm. + smoothing	Bite vs. non-bite HMM	DNN
Microm. accuracy	78.8 %	78.8 %	78.8 %	80.0 %
Bite vs. non-bite precision	0.44	0.77	0.62	0.91
Bite vs. non-bite recall	0.93	0.73	0.64	0.93
Bite vs. non-bite F-measure	0.59	0.75	0.63	0.91
Bite count relative error	110.7 %	6.9 %	8.7 %	10.7 %

Table 1: Accuracy of micromovement and bite vs. non-bite recognition.

6. CONCLUSION

In this paper we presented three approaches for bite (or food intake) counting using sensors in a smart watch. Each of them consisted of two main steps: the recognition of bite-related micromovements and the recognition of the actual bites based on that. Classical and CNN-based micromovement recognition proved comparable. The approach for bite detection based on HMM smoothing proved best in terms of the number of miscounted bites, while the DNN-based approach proved best in terms of precision and recall (and comparable to the state of the art). The contribution of the paper is the novel approach based on HMM smoothing, and its comparison with the two other main approaches known from the literature.

Since the success of the approach based on HMM smoothing was heavily dependent on its parameter settings, we decided to integrate the DNN-based method in the WellCo system. Our main task for the future is to merge the recognised bites into meals, and associate each meal with an estimate of the amount of food eaten, since this is what the WellCo virtual coach needs.

7. ACKNOWLEDGMENTS

We thank Matej Sudac for his work on classical micromovement recognition. The WellCo project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 769765.

8. REFERENCES

- [1] Reščič, N., Valenčič, E., Mlinarič, E., Barbara Koroušič Seljak, B., and Luštrek, M. 2019. Mobile nutrition monitoring for well-being. To appear in *UbiComp conference, WellComp workshop*.
- [2] Champagne, C., Bray, G., Monteiro, J., Tucker, E., and Volaufovaand, J. 2002. Energy intake and energy expenditure: A controlled study comparing dietitians and non-dietitians. *Journal of the American Dietetic Association*, 102, 10, 1428–1432.
- [3] Mezgec, S., and Koroušič Seljak, B. 2017. NutriNet: A deep learning food and drink image recognition system for dietary assessment. *Nutrients* 9, 7, 657.
- [4] Dong, Y., Scisco, J., Wilson, M., Muth, E., and Hoover, A. 2014. Detecting periods of eating during free-living by tracking wrist motion. *Journal of Biomedical and Health Informatics* 18, 4, 1253–1260.
- [5] Dong, Y., Hoover, A., Scisco, J., and Muth, E. 2012. A new method for measuring meal intake in humans via automated wrist motion tracking. *Applied Psychophysiology and Biofeedback* 37, 3, 205–215.
- [6] Liu, J., Johns, E., Atallah, L., Pettitt, C., Lo, B., Frost, G., and Yang, G.-Z. 2012. An intelligent food-intake monitoring system using wearable sensors. In *2012 9th Intl. Conference on Wearable and Implantable Body Sensor Network*.
- [7] Hosseini, A., Kalantarian, H., and Sarrafzadeh, M. 2016. Adaptive data processing for real-time nutrition monitoring. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*.
- [8] Alshurafa, N., Kalantarian, H., Pourhomayoun, M., Liu, J. J., Sarin, S., and Shahbazi, B. 2015. Recognition of nutrition intake using time-frequency decomposition in a wearable necklace using a piezoelectric sensor. *IEEE Sensors Journal* 15, 7, 3909–3916.
- [9] Kyritsis, K., Lefkothea Tatli, C., Diou, C., and Delopoulos, A. 2017. Automated analysis of in meal eating behavior using a commercial wristband IMU sensor. In *Proceedings of 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*.
- [10] Kyritsis, K., Diou, C., and Delopoulos, A., 2019. Modeling wrist micromovements to measure in-meal wating behavior from inertial sensor data. *IEEE Journal of Biomedical and Health Informatics*.
- [11] Cvetković, B., Drobnič, V., and Luštrek, M. 2017. Recognizing hand-specific activities with a smartwatch placed on dominant or non-dominant wrist. In *Information Society (IS) conference*.
- [12] Cvetković, B., Szeklicki, R., Janko, V., Lutomski, P., and Luštrek, M. 2018. Real-time activity monitoring with a wristband and a smartphone. *Information Fusion* 43, 77–93.

Austrian-Slovenian Intelligent Tourist-Information Center Platform

Jernej Zupančič
Jožef Stefan Institute and
Jožef Stefan International
Postgraduate School
Jamova cesta 39
Ljubljana, Slovenia
jernej.zupancic@ijs.si

Gregor Grasselli
Jožef Stefan Institute and
Jožef Stefan International
Postgraduate School
Jamova cesta 39
Ljubljana, Slovenia
gregor.grasselli@ijs.si

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

ABSTRACT

We present the AS-IT-IC platform – Austrian-Slovenian intelligent tourist-information platform that enables live communication between tourist workers and tourists through a simple chat widget embedded on the existing tourism website, and includes TOUR-BOT, a virtual tourist assistant, for cases when no tourist worker is available. TOUR-BOT can communicate in natural language in English, German and Slovene. The AS-IT-IC platform enables trip planning by providing: attraction search in natural language, trip manipulation using natural language and traditional interaction through buttons, and optimal route calculation between selected trip points. Additionally, the AS-IT-IC platform enables attraction data management for tourist workers and data update suggestions by anonymous users.

Keywords

tourist-information system, virtual assistants, conversational platform

1. INTRODUCTION

Currently, the common way to obtain relevant information [5, 2] about cultural and natural heritage sights and plan a trip is through user-unfriendly web search and hard-to-find websites, usually managed at the government or local-authority levels. After finding relevant information, the trip is planned using route finder such as Google Maps¹ or more advanced tour planners such as e-Turist² [1] or TripHobo³.

However, a new solution for tourists visiting Slovenia and Austria is emerging. Within the AS-IT-IC project [7], accepted in the cross-border Cooperation Programme Interreg V-A Slovenia-Austria in the programme period 2014-2020, a novel information communication technologies (ICT) system was implemented that enables several features for empowering the tourist:

1. Live-chat with the tourist information providers (tourist service providers, tourist offices, municipalities, and citizens).
2. Tour planing and tourism information search in natural language conversation with the virtual assistant.

¹<https://www.google.si/maps>

²<http://e-turist.si>

³<https://www.triphobo.com>

3. Attraction information suggestion.

The designed ICT tools were integrated into the AS-IT-IC Platform and are accessible from the project homepage⁴ through the tourist information provider access point⁵ and the tourist access point⁶.

The rest of the paper is structured as follows. In Section 2 an example of the platform usage is provided. In Section 3 the platform architecture and platform components are described and Section 4 concludes the paper.

2. USE CASE

To demonstrate the AS-IT-IC platform idea, consider the following example. Imagine Mary, a tourist worker who wants to enable live-chat feature on her website <https://visit-hidden-slovenia.com>. After registering on the AS-IT-IC platform, copy-pasting only a few lines of HTML and Javascript code into her website template and doing some color and text customization if so desired, she has a tourist live-chat widget embedded into her website. When John, a tourist, comes looking for information on her website, he clicks on the live-chat widget button to start a conversation. First, a virtual assistant greets him to kick-start the dialogue. John then searches for particular categories in the limited area such as: “horseriding near Bovec”, “wine cellars in Nova Gorica”, “adrenaline sports near Soča river”, “accommodation in Goriška Brda” etc. John picks some of the results and includes them into his itinerary.

During the interaction, the tourist can communicate with logged-in humans or with a virtual assistant, the later most useful when no human is on-line, e.g., at night. In our example, John invites Mary into the conversation and they chat about the transit options and other questions the tourist may have. While chatting, the tourist worker has full overview of the conversation the tourist had with the virtual assistant and the currently chosen itinerary. This helps Mary provide relevant answers quickly. In the case that Mary does not know the answer, she can simply invite another tourist worker into the conversation in order to collaborate in providing all the information the tourist needs.

⁴<https://as-it-ic.ijs.si>

⁵<https://asitic.docker-e9.ijs.si>

⁶<https://asitic-chat-api-frontend.docker-e9.ijs.si/asitic-app.html>

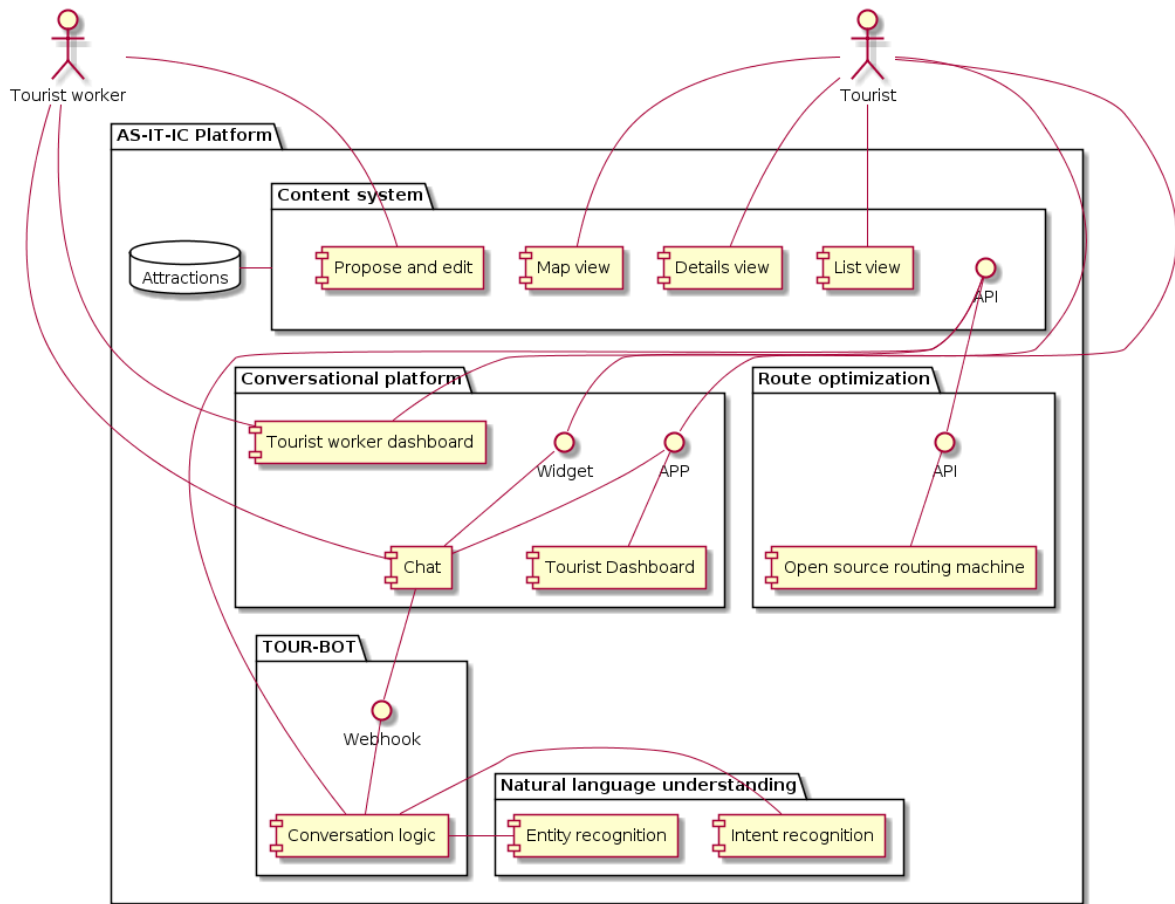


Figure 1: AS-IT-IC platform architecture overview.

3. ARCHITECTURE

The AS-IT-IC platform (Fig. 1) comprises several modules that enable the features mentioned in Sections 1 and 2. The main modules are:

1. *Conversational platform* that enables real-time communication between users and lays out the communication infrastructure.
2. *TOUR-BOT*, a virtual assistant that is able to hold a conversation with tourists by understanding the natural text entered by the user, responding to button clicks within the context, and performing actions with regards to third party systems.
3. *Natural language understanding*, a module that transforms unstructured text into structured information by recognizing entities and intent of the received text.
4. *Content system* enables to create, read, update and delete operations regarding the points-of-interest (POI) available on the AS-IT-IC platform. Additionally, it deals with the information presentation of specific POI or a group of POIs.
5. *Route optimization*, a service for finding optimal route between a set of points.

3.1 Conversation Platform

The conversation platform is based upon the popular open-source software Rocket.Chat⁷, which provides a full-featured modern chat application by: enabling virtual assistant integration through webhooks, enabling the home page customization, allowing for the user interface localization, providing responsive interface for desktop and mobile screen sizes, enabling iframe integration through message posting. There are three entry points to the conversational platform:

1. Rocket.Chat app itself; meant for access by the tourist workers.
2. AS-IT-IC app: a full screen application with AS-IT-IC-specific menu and the embedded Rocket.Chat app; meant for access by frequent tourist users.
3. AS-IT-IC widget: a floating widget, which can be embedded into an existing website. It includes AS-IT-IC-specific menu and embedded Rocket.Chat app; meant for access by an anonymous tourist user visiting a tourism website.

Since two types of users are expected for the AS-IT-IC platform, the conversation platform has to enable different home-

⁷<https://rocket.chat>

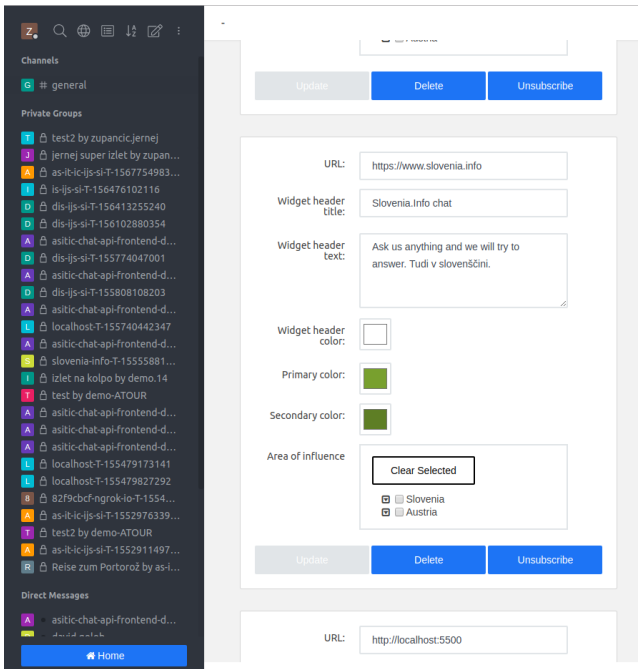


Figure 2: Tourist worker dashboard view.

page views, which are based on the user roles. To the tourist user, only the information regarding the planned trips is shown. This way, every planned trip is easily accessible from the home page. For the tourist workers, the configuration options for the AS-IT-IC widget deployed on the tourist workers' website are available (Fig. 2). This enables the tourist worker to customize the widget's appearance and converse with tourists from one central location.

3.2 Virtual Tourist Assistant

Our virtual tourist assistant "TOUR-BOT" can provide information to a tourist when no tourist worker is available or when the tourist prefers a robotic assistant (Fig. 3). TOUR-BOT has access to the conversation and the conversation context, which holds structured information about the current conversation. It responds to the users' button clicks and text inputs in natural language. In order to process the natural language it utilizes the "Natural language understanding" service, which analyzes the text and returns structured information about the text, consisting of:

1. Text intent – what does the user want to do; an example: the intent of "I want to go to Planica" is **obtain direction**), and
2. Recognized entities – what points-of-interest are mentioned in the text; an example: the entity in "I want to go to Planica" is **Planica**).

TOUR-BOT stores the conversation dialog flow rules, which enables it to hold a meaningful conversation that spans several tourist – TOUR-BOT interactions. For instance, when the tourist states "I want to go to Planica", TOUR-BOT asks "How will you visit those places" and offers different modes of

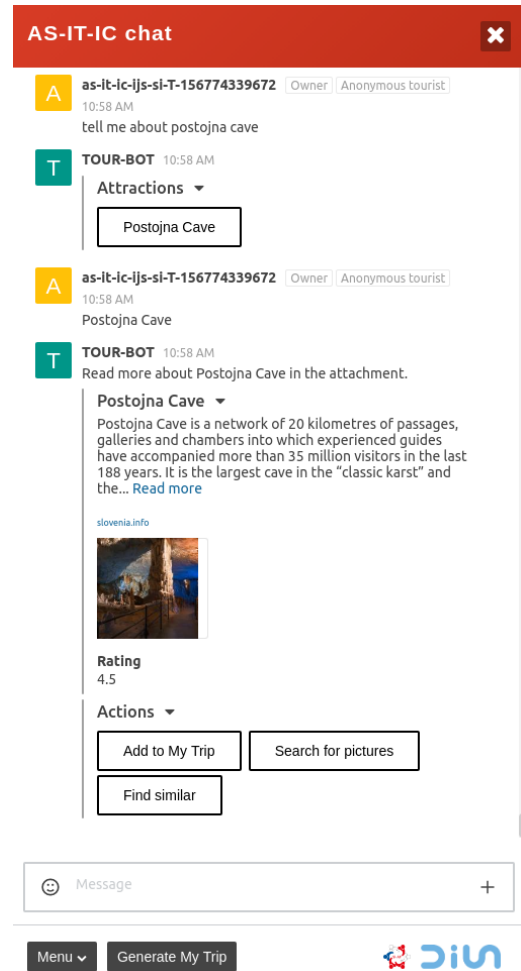


Figure 3: Tourist to TOUR-BOT chat example.

transport. After the tourist provides feedback, proper links to Google Maps⁸ or Openroute service⁹ are generated and displayed. The following intents are currently supported by TOUR-BOT:

1. General POI search. Examples: "Tell me about Postojna cave", "horseriding near Soča river", "accommodation near Ljubljana castle".
2. Path to attraction request. An example: "I want to go to Ljubljana castle".
3. Similar search request. An example: after reading about the Ljubljana castle – "anything similar nearby".
4. Trip manipulation actions: add a POI to the trip, clear the trip, generate route for the trip. Examples: after reading about the Ljubljana castle – "add this to my trip", "clear my trip", "generate route" etc.
5. Small talk intents: greeting, critique, about, help, praise and others. Examples: "hi", "who are you", "you are bad", "help me", "you are great" etc.

⁸<https://www.google.com/maps>

⁹<https://openrouteservice.org>

3.3 Natural Language Understanding

The natural language understanding service comprises two parts: entity recognition and intent recognition. Entity recognition is available for entities that were obtained from tourism websites and other sources. See [8] for more information. In summary, 20,999 entities are available in the system, of which 8,734 are attractions and others are mostly geographical entities. The entity recognition is based on the probability that an entity is present in the text. The probability is estimated on word distances between the entity words and the text input words. Entity recognition is further explained in [6].

Intent recognition is a text classification task. Based on the text representation, a machine learning model is built to best classify the text in the training examples. While different processing and classification pipelines were tested, the best performing pipeline is the following: First, the input text is tokenized¹⁰. Second, the stopwords are removed from the text. Third, word vector (using FastText embeddings [3] and the pymagnitude package [4]) for each remaining token is queried. Mean vector is computed and used as the text embedding representation. Support vector machine from the scikit-learn project¹¹ is then used to obtain the machine learning model. Using this pipeline we were able to achieve F1 score of about 0.8 on our tourist queries dataset that comprised about 100 queries.

3.4 Content System

The content system stores all the POI data [8] used across the AS-IT-IC platform. It enables tourist workers to create, view, update and delete specific POI information. Additionally, it enables crowd-sourcing the data-gathering by allowing anonymous users to provide suggestions for updating POI information. The suggested update is publicly visible only when the user with sufficient rights (e.g., tourist worker) approves the suggestion. The content system is further responsible for detailed POI presentation and the overview presentation of all POIs within the system, either in the form of a map or an interactive list¹².

3.5 Route Optimization

Route optimization enables finding quickest route for visiting a set of chosen points, while taking into account the preferred mode of transport and the road infrastructure within some region. We utilized the Open Source Routing Machine¹³ (OSRM) for data preparation for the regions of Slovenia and Austria. Further, the OSRM `trip` endpoint was used to solve the Traveling Salesman Problem, which is solved “using a greedy heuristic (farthest-insertion algorithm) for 10 or more waypoints and using brute force for less than 10 waypoints”¹⁴.

4. CONCLUSIONS

In this paper we presented an intelligent tourist-information platform for Austria and Slovenia as part of the AS-IT-IC

project. The platform enables live communication between tourist workers and tourists. Additionally, tourists can converse with TOUR-BOT, a virtual assistant, in natural language in Slovene, German or English in order to search for relevant attractions and plan a trip. Using the AS-IT-IC platform, a tourist can easily plan a visit to a specific region, while tourist workers can establish direct communication with the tourists and obtain live feedback on their interests. In summary, the AS-IT-IC tourist platform enables several novel user-friendly functions for tourists and tourist workers.

5. ACKNOWLEDGMENTS

This work was co-financed by Cooperation Programme Interreg V-A Slovenia-Austria 2014-2020, project AS-IT-IC.

6. REFERENCES

- [1] B. Cvetković, H. Gjoreski, V. Janko, B. Kaluža, A. Gradišek, M. Luštrek, I. Jurinčič, A. Gosar, S. Kerma, and G. Balazič. e-Turist: An intelligent personalised trip guide. *Informatica*, 40(4):447, 2016.
- [2] G. Grasselli and J. Zupančič. Tourism related ICT tools: A review. In *Proceedings of the 21st International Multiconference Information Society - IS 2018*, volume E, pages 10–13, Ljubljana, Slovenia, 2018. Jožef Stefan Institute.
- [3] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.
- [4] A. Patel, A. Sands, C. Callison-Burch, and M. Apidianaki. Magnitude: A fast, efficient universal vector embedding utility package. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 120–126, 2018.
- [5] B. Peischl, O. A. Tazl, and F. Wotawa. Open questions of technology usage in the field of tourism. In *Proceedings of the 20th International Multiconference Information Society - IS 2017*, volume E, pages 29–22, Ljubljana, Slovenia, 2017. Jožef Stefan Institute.
- [6] M. Štravs and J. Zupančič. Named entity recognition using gazetteer of hierarchical entities. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pages 768–776. Springer, 2019.
- [7] J. Zupančič and M. Gams. Austrian-Slovenian intelligent tourist information center: Project progress report. In *Proceedings of the 21st International Multiconference Information Society - IS 2018*, volume E, pages 5–9, Ljubljana, Slovenia, 2018. Jožef Stefan Institute.
- [8] J. Zupančič, O. A. Tazl, B. Mahnič, and G. Grasselli. AS-IT-IC databases. In *Proceedings of the 21st International Multiconference Information Society - IS 2018*, volume E, pages 14–17, Ljubljana, Slovenia, 2018. Jožef Stefan Institute.

¹⁰<https://repo.ijs.si/DIS-AGENTS/reldi-tokeniser>

¹¹<https://scikit-learn.org>

¹²<https://eturist.docker-e9.ijs.si>

¹³<http://project-osrm.org>

¹⁴<http://project-osrm.org/docs/v5.22.0/api/#trip-service>

Razvoj postopka diarizacije govorcev z algoritmi strojnega učenja

Marko Katrašnik
Institut „Jožef Stefan“
Jamova cesta 39
1000 Ljubljana, Slovenia
marko.katrasnik@gmail.com

Junoš Lukan, Mitja
Luštrek
Institut „Jožef Stefan“
Mednarodna podiplomska šola
Jožefa Stefana
Jamova cesta 39
1000 Ljubljana, Slovenia
{junos.lukan,
mitja.lustrek}@ijs.si

Vitomir Štruc
Fakulteta za elektrotehniko,
Univerza v Ljubljani
Tržaška cesta 25
1000 Ljubljana, Slovenia
vitomir.struc@fe.uni-lj.si

POVZETEK

Pomemben del konteksta uporabnikov mobilnih telefonov so njihove žive socialne interakcije. Zaznamo jih lahko s pomočjo mikrofona, pri čemer lahko zaznamo prisotnost človeškega govora, ugotavljamo število govorcev in določamo, kdaj je govoril kateri od govorcev, čemur pravimo diarizacija. V članku sta predstavljena detektor govora in širši postopek diarizacije govorcev, za katera smo uporabili že obstoječa orodja in jih prilagodili za široko uporabnost na posnetkih z mobilnih telefonov. Za zaznavo govora smo uporabili logistično regresijo, ki je računsko nezahteven algoritem, hkrati pa je imel visoko točnost, skoraj 90 %, v različnih akustičnih okoljih. Za razvoj ostalih korakov diarizacije smo uporabili že obstoječe zbirke podatkov, posneli pa smo tudi majhno lastno zbirko. Za snemanje smo uporabili telefone različnih proizvajalcev in na ta način preverili robustnost našega postopka v primerjavi z že razvitimi metodami zaznavanja govora in diarizacije govorcev. V kontroliranih pogojih je postopek deloval primerljivo z že obstoječimi, na posnetkih iz vsakdanjega delovnega okolja pa je dosegel izrazito boljše rezultate.

1. UVOD

V okviru projekta na temo stresa na delovnem mestu (angl. Stress At Work project, StrAW) [3] želimo analizirati in opisati odnose med izkušnjami psihosocialnega stresa v delovnem okolju, vsakdanjimi aktivnostmi in dogodki na delu in fiziološkimi signali [16] ter vedenjskimi vzorci, ki jih lahko zaznamo avtomatsko s pomočjo tehnologije. Eden od pomembnih virov podatkov, ki jih lahko uporabimo v ta namen so senzorski podatki in spremljanje interakcije z mobilnim telefonom. S temi podatki je mogoče prepoznati pomemben del uporabnikovega konteksta, kot so socialne interakcije in pogovori. Tega problema smo se lotili v pričujočem delu in že objavljeni diplomski nalogi [11].

V prispevku je predstavljen postopek za diarizacijo govorcev. To je proces označevanja posnetka z informacijo o tem, kateremu govorcu pripadajo določeni segmenti v posnetku. V splošnem zajema tri glavne korake: zaznavanje človeškega govora, iskanje mej med deli posnetka, med katerimi so govorili različni govorci, ter združevanje teh segmentov glede na identiteto govorca. Ker smo želeli večji nadzor nad delovanjem postopka v različnih akustičnih pogojih, smo se prvi

komponenti, detekciji govora, posvetili ločeno od ostalih.

Za zaznavanje govora se najpogosteje uporablja pristop z mešanicami Gaussovih porazdelitev (angl. Gaussian mixture models, GMM) [17], za optimizacijo telefonskih klicev preko internetnega protokola pa je standardiziran pristop z uporabo statističnega modeliranja [2]. V zadnjem času se za ta namen pogosto uporabljajo tudi nevronske mreže (npr. [10]).

Diarizacija govorcev se v literaturi (dober pregled je v [1]) največkrat izvaja na posnetkih novic in sestankov, pri čemer imajo uporabljeni posnetki ugodne akustične značilnosti. Pri tem nekateri pristopi izkoristijo snemanje z več mikrofoni in s pomočjo različnih zvočnih tokov izboljšajo točnost določanja govorcev.

Cilj našega dela je bil prilagoditev postopka za uporabo na posnetkih iz vsakdanjega življenja, pri katerih bi za snemanje uporabili mikrofona mobilnega telefona.

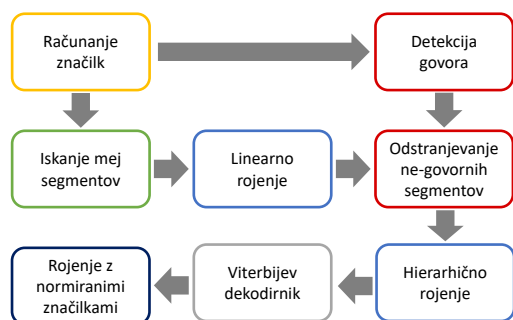
2. METODE

2.1 Postopek diarizacije govorcev

Za zaznavanje govora (angl. voice activity detection, VAD) so uporabne tako značilke v časovni (na primer najvišje in najnižje vrednosti signala) kot v frekvenčni domeni (na primer ploščatost spektra). Poleg teh se za opis človeškega glasu uporabljajo tudi specializirane značilke, kot so kepstralni koeficienti melodičnega spektra (angl. mel-frequency cepstral coefficients, MFCC) in zaznavni kepstralni koeficienti linearne napovedi z relativno spektralno transformacijo (angl. RASTA perceptual linear prediction coefficients, RASTA PLP-CC) [8]. Obe vrsti značilk delujeta v kepstralni domeni, v katero pridemo z inverzno Fourierovo transformacijo logaritma spektra signala, obenem pa z različnimi transformacijami poskušajo posnemati odziv človeškega sluha na zvok.

V našem postopku diarizacije govorcev, prikazanem na sliki 1, smo najprej izračunali značilke, ki smo jih uporabili v ostalih komponentah diarizacije govorcev. Izračunu značilk je sledilo iskanje mej med segmenti govorcev. Združevanje segmentov istega govorca se v procesu diarizacije ponovi večkrat na različne načine (modro obrobjeni koraki na sliki 1). Najprej smo z linearnim rojenjem (angl. linear clustering) združili le zaporedne segmente, ki pripadajo istemu govorcu. Nato smo na podlagi detektorja govora, ki je bil predstavljen v

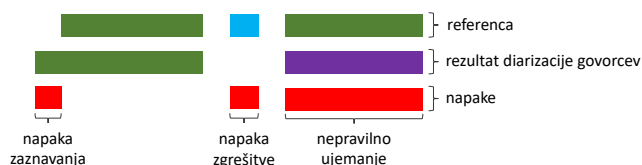
prejšnjem koraku, odstranili ne-govorne segmente in s hierarhičnim rojenjem že združili večino, tudi nezaporednih, segmentov istega govorca. Viterbijev dekodirnik dodatno izboljša začetno segmentacijo s pomočjo prikritih modelov Markova. Zadnji korak pa je bilo hierarhično rojenje z normiranimi značilkami, ki združi še roje istih govorcev, ki so bili do tega koraka ločeni zaradi različnega akustičnega ozadja.



Slika 1: Koraki v postopku diarizacije govorcev.

2.2 Vrednotenje rezultatov

Zaznavanje govora smo vrednotili s pomočjo ustaljenih mer: točnosti (angl. accuracy) in priklica (angl. recall). Poleg teh dveh mer pa smo za vrednotenje diarizacije uporabljali tudi mero napake med ujemanjem referenčnih in samodejno pridobljenih in označenih segmentov, krajše mero DER (angl. diarization error rate). Izračunamo jo kot delež napačno klasificiranih delov posnetka, kjer so možne tri različne napake (slika 2): napaka zaznavanja (ne-govor označen kot govor), napaka zgrešitve (govor označen kot ne-govor) in nepravilno ujemanje (klasificiran napačen govor). Od običajne točnosti se razlikuje po tem, da delež izračunamo le glede na dele posnetka, v katerih je bil dejansko prisoten govor.



Slika 2: Pri vrednotenju diarizacije upoštevamo tri različne vrste napak. Njihov časovni delež v delih posnetka, v katerih je prisoten govor, je mera DER.

3. REZULTATI

3.1 Orodja

Orodje „openSMILE“ [9] omogoča izračun vseh vrst akustičnih značilk, omenjenih v prejšnjem podpoglavju. V našem delu smo za njihov izračun uporabili okna z dolžino 25 ms in korakom 10 ms. Omeniti velja, da „openSMILE“ ponuja tudi že prednaučeno povratno nevronska mrežo z dolgim kratkoročnim spominom (angl. long short-term memory recurrent neural network, LSTM-RNN) za detekcijo človeškega glasu. Izhod te nevronske mreže smo pri našem detektorju govora uporabili kot značilko in kot referenco pri primerjavi točnosti drugih algoritmov za zaznavanje govora.

Glavni del diarizacije smo opravili z orodjem „LIUM SpkDiarization“ [14]. Poleg prilagoditve parametrov tega orodja pa smo zamenjali prvo komponento zaznavanja govora, ki smo jo implementirali s pomočjo orodja za strojno učenje „Weka“ [6].

3.2 Podatkovne zbirke

Pri razvoju postopka diarizacije govorcev smo uporabljali tri podatkovne zbirke: eno za učenje in validacijo zaznavanja govora, drugo za nastavljanje hiperparametrov ostalih komponent diarizacije in tretjo za končno testiranje postopka in primerjavo z drugimi metodami.

Kot učno množico za prvi korak zaznavanja govora smo uporabili „VAD-toolkit“ [12]. Gre za pogovore dveh korejskih govorcev, posnetih s telefonom Samsung Galaxy S8 v štirih različnih okoljih z različnim hrupom v ozadju: v sobi, v parku, na avtobusni postaji in na gradbišču. Posnetki so označeni z deli govora in ne-govora (binarne oznake), pri čemer govor vsebuje približno tretjina celotnega trajanja posnetkov.

Za razvoj ostalih komponent diarizacije smo uporabljali podatkovno množico „AMI Corpus“ [5], ki vsebuje označene posnetke sestankov v angleškem jeziku. Izbrali smo tri posnetke, v katerih je sodelovalo od 3 do 5 govorcev s trajanjem od 19 min do 66 min. Da bi posneli različne akustične pogoje, smo jim umetno dodali šum s pomočjo orodja „Audio Degradation Toolbox“ [13]. Dodali smo jim posnetek z ulice z razmerjem med signalom in šumom 15 dB in posnetek iz gostilne z razmerjem 20 dB.

Kot testno množico za ovrednotenje celotnega postopka diarizacije in primerjavo z drugimi obstoječimi orodji smo uporabili drugo podmnožico zbirke „AMI Corpus“, dodatnih 12 posnetkov. Poleg tega smo posneli še tri lastne posnetke sestankov oziroma pogovora, za kar smo uporabili štiri različne pametne telefone: Huawei P Smart, Motorola Moto X, Samsung Galaxy S6 in Nokia 6.

3.3 Zaznavanje govora

Za zaznavanje govora smo najprej izbrali najbolj primeren algoritem strojnega učenja. Algoritme, ki smo jih preizkusili, prikazuje tabela 1 in so natančneje opisani v [6]. Z vidika točnosti se je najbolje izkazala logistična regresija. Kljub temu, da imajo nekatere druge metode za govor boljši priklic, smo v nadaljevanju izbrali ravno to metodo, saj je tudi relativno računsko nezahtevna.

Tabela 1: Primerjava uspešnosti različnih algoritmov za detekcijo govora pri uporabi vseh izračunanih značilk. Prikazani so priklic za razreda govora ($priklic_g$) in ne-govora ($priklic_{ng}$) ter skupna točnost.

Algoritem	$priklic_g$	$priklic_{ng}$	točnost
Logistična regresija	93,7	97,1	96,1
SVM	94,4	96,8	96,1
Večplastni perceptron	95,6	95,5	95,5
Naključni gozd	93,8	95,4	94,8
AdaBoost	93,0	94,4	94,1
KNN	87,1	93,6	91,4
J48	90,0	91,1	90,7
Naivni Bayes	60,8	93,6	84,1

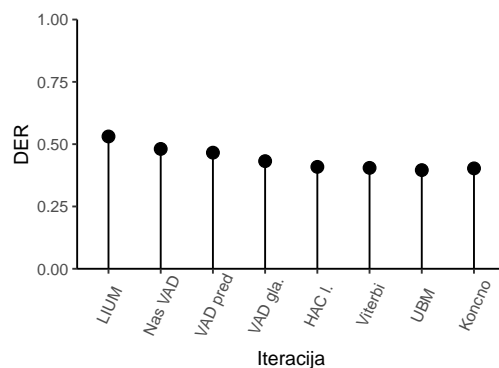
V naslednjem koraku smo izbrali najboljše značilke, saj smo sprva uporabljali vse smiselne, ki jih lahko izračunamo z orod-

jem „openSMILE“. V ta namen smo uporabili več metod: izboljššan postopek Relief (angl. ReliefF), izbor na podlagi korelacij (angl. correlation feature selection, CFS) in metodo z ovojnico (angl. wrapper) [6]. S tem smo nabor značilk zmanjšali s 104 na 54 značilk. Ohranili smo najvišjo, najnižjo in absolutno najvišjo vrednost oknjenega signala, ploščatost spektra, 13 značilk MFCC, 18 RASTA PLP-CC koeficientov in njihove koeficiente delta regresije ter izhod povratne nevronske mreže. Pri tem je točnost logistične regresije ostala enaka, priklic govora pa se je nekoliko izboljšal.

Kot omenjeno, smo izhod nevronske mreže pri našem detektorju govora uporabili kot značilko. Izkazalo se je, da je to v našem postopku najpomembnejša značilka, vendar vključitev ostalih poveča točnost s 86,6 % na 96,1 %.

3.4 Diarizacija govorcev

Komponente diarizacije govorcev smo spreminjali iterativno. Začeli smo z osnovnim postopkom LIUM, zamenjali detektor govora za opisanega v podpoglavju 3.3, nato pa spreminjali še hiperparametre. Slika 3 prikazuje, kako se je spreminjala napaka DER z zaporednimi spremembami postopka diarizacije.



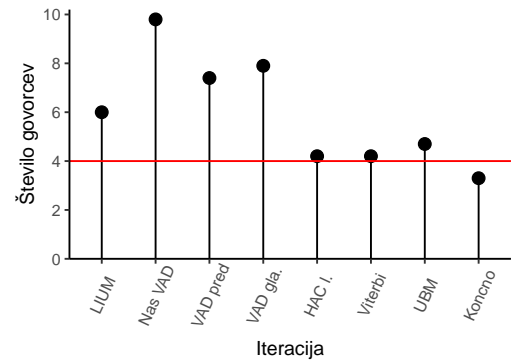
Slika 3: Spreminjanje napake DER skozi iterativne izboljšave postopka diarizacije. Največja, petodstotna, sprememba (označena z „Nas VAD“) je posledica zamenjave privzetega detektorja govora za lastnega.

V splošnem smo izbrali spremembe, ki so postopek izboljšale, tako da so zmanjšale mero DER. Največjo izboljšavo smo dosegli z zamenjavo detektorja govora („Nas VAD“), dodatno smo izboljšali delovanje postopka z odstranjevanjem negovornih segmentov v zgodnejšem koraku („VAD pred“) in nastavitvijo parametrov glajenja rezultatov detektorja govora („VAD gla.“).

Drugi korak (iteracija, označena s „HAC I.“), ki se je izkazal za pomembnega, je bila sprememba parametra v hierarhičnem rojenju (angl. hierarchical agglomerative clustering, HAC). Rojenje smo naredili bolj agresivno, s tem pa zmanjšali končno število rojev oziroma govorcev. Učinek te spremembe je še bolj izrazito opazen na sliki 4, kjer se je število govorcev močno približalo pravemu.

Nato smo prilagodili parametre Viterbijevega dekodirnika, namen katerega je, da izboljša začetno segmentacijo.

V zadnjih dveh korakih, označenih z „UBM“ in „Končno“, smo spremenili še uporabo prednaučenega splošnega modela



Slika 4: Število zaznanih govorcev se je spreminjalo z vsako iteracijo postopka diarizacije.

govora (angl. universal background model, UBM). Ta model se uporablja za naknadni postopek rojenja, vendar bi ga bilo treba prilagoditi na uporabljeni podatkovni množici. Namesto tega smo komponento nadomestili s preprostejšim hierarhičnim rojenjem, na račun povečanja splošnosti pa se je z zadnjo spremembo nekoliko poslabšala napaka DER (slika 3) in napaka ocene števila govorcev (slika 4).

3.5 Primerjava rezultatov

Končno različico postopka, ki smo ga razvili, smo primerjali z nespremenjeno metodo LIUM ter drugo, v literaturi pogosto uporabljeno metodo diarizacije ALIZE [4].

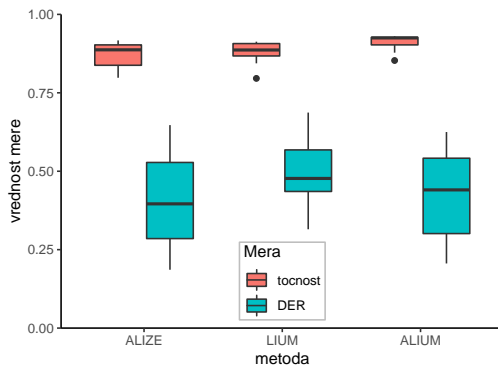
Slika 5 prikazuje točnost detektorja govora in napako DER našega postopka diarizacije (označenega z „ALIUM“) v primerjavi z drugima dvema postopkoma iz literature na posnetkih iz „AMI Corpus“. S spremembami, opisanimi v prejšnjem poglavju, smo izboljšali točnost zaznavanja govora: kot izračunano z analizo variance, se metode razlikujejo statistično pomembno ($\chi^2(2) = 32,8, p < 0,001$). Pri tem se je točnost izboljšala predvsem zaradi priklica razreda govora, medtem ko je priklic razreda ne-govora dejansko slabši kot pri postopku ALIZE. Po drugi strani se metode razlikujejo tudi v napaki DER ($\chi^2(2) = 14,8, p < 0,001$), vendar je bil glede na to mero najboljši postopek ALIZE.

Iste tri metode smo primerjali tudi na treh lastnih posnetkih. V tem primeru se je naš postopek obnesel mnogo bolje tudi z vidika napake DER, saj je napaka v povprečju znašala DER = 0,341, medtem ko je bila pri nespremenjenem postopku LIUM DER = 0,828 in ALIZE DER = 0,794.

Delovanje metode diarizacije smo primerjali tudi preko posnetkov z različnih mobilnih telefonov. Točnost zaznave govora se je med posnetki z različnih telefonov razlikovala za 1 %, napaka DER pa za največ 3,5 %.

4. ZAKLJUČKI

Naš postopek diarizacije govorcev, še posebej pa korak zaznavanja govora, deluje zanesljivo na raznovrstnih posnetkih. Že obstoječe metode v dobro kontroliranih pogojih in sodeč po nekaterih merah sicer dajejo nekoliko boljše rezultate, vendar kaže, da je v našem delu prilagojen postopek širše uporaben. To se je pokazalo v doslednih rezultatih na različnih zbirkah podatkov, in mnogo boljšimi rezultati na zbirki, posneti z mobilnimi telefoni različnih proizvajalcev.



Slika 5: Točnost detektorja govora, opisanega v podpoglavju 3.3, in napaka DER pri diarizaciji s postopkom, opisanim v podpoglavju 3.4 in označenim z „ALIUM“. Ti dve meri smo primerjali z originalnim LIUM postopkom ter postopkom ALIZE na dvanajstih posnetkih „AMI Corpus“.

Druga pomembna prednost razvitega postopka je njena nizka računrska zahtevnost. Logistična regresija je preprost algoritem, zaradi česar je detekcijo govora mogoče izvajati v realnem času na sodobnih mobilnih napravah z operacijskim sistemom Android. Večina računrske zahtevnosti tako prinesejo nadaljnji koraki diarizacije: čas računanja je linearno naraščal z dolžino posnetka, čas izvedbe celotnega postopka pa je bil na prenosnem računalniku v povprečju 20-krat krajši od trajanja posnetka.

Na posnetkih iz „AMI Corpus“ se je metoda ALIZE brez prilagoditev glede na napako DER izkazala bolje od naše (slika 3). Ti posnetki imajo v primerjavi s posnetki s telefonov iz vsakdanjega življenja ugodno razmerje med signalom in šumom in tudi večjo glasnost. Naš postopek bi tako morda lahko izboljšali s predobdelavo posnetkov, predvsem z normiranjem glasnosti in odstranjevanjem šuma. Druga izboljšava, ki je bila v literaturi že preizkušena [7], je iskanje mej med segmenti govorcev od začetka do konca posnetka in nato še v nasprotni smeri.

Za implementacijo postopka za samodejno zaznavo pogovorov s pomočjo pametnih telefonov pa bomo v prihodnje pozornost poleg točnosti namenili tudi porabi energije. Kljub računrski preprostosti detektorja govora je namreč že samo snemanje z mikrofonom veliko breme za baterijo. Govor je mogoče zaznati tudi brez neprestanega snemanja, na primer z uporabo adaptivnega vzorčenja [15].

Literatura

- [1] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals. Speaker diarization: A review of recent research. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(2):356–370, 2012.
- [2] A. Benyassine, E. Shlomot, H.-Y. Su, D. Massaloux, C. Lamblin, and J.-P. Petit. ITU-T recommendation G. 729 Annex B: A silence compression scheme for use with G. 729 optimized for V. 70 digital simultaneous voice and data applications. *IEEE Communications Magazine*, 35(9):64–73, 1997.
- [3] L. Bolliger, J. Lukan, M. Luštrek, D. D. Bacquer, and E. Clays. Disentangling the sources and context of daily work stress: Study protocol of a comprehensive realtime modelling study using portable devices. 2018. Employability 21.
- [4] J.-F. Bonastre, F. Wils, and S. Meignier. ALIZE, a free toolkit for speaker recognition. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, volume 1, pages I/737–I/740. IEEE, 2005.
- [5] J. Carletta. Announcing the AMI meeting corpus. *The ELRA Newsletter*, 11(1):3–5, 2006.
- [6] F. Eibe, M. A. Hall, and I. H. Witten. *The WEKA Workbench. Online Appendix for “Data Mining: Practical Machine Learning Tools and Techniques”*, fourth edition, 2016.
- [7] E. El-Khoury, C. Senac, and J. Piquier. Improved speaker diarization system for meetings. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4097–4100. IEEE, 2009.
- [8] F. Eyben. *Real-time Speech and Music Classification by Large Audio Feature Space Extraction*. Springer International Publishing, 2016.
- [9] F. Eyben, F. Weninger, F. Gross, and B. Schuller. Recent developments in openSMILE, the Munich open-source multimedia feature extractor. In *Proceedings of the 21st ACM international conference on Multimedia*, pages 835–838. ACM, 2013.
- [10] T. Hughes and K. Mierle. Recurrent neural networks for voice activity detection. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 7378–7382. IEEE, 2013.
- [11] M. Katrašnik. *Razvoj postopka diarizacije govorcev z algoritmi strojnega učenja*. Diplomsko delo, Univerza v Ljubljani, Fakulteta za elektrotehniko, Fakulteta za računalništvo in informatiko, 2019.
- [12] J. Kim and M. Hahn. Voice activity detection using an adaptive context attention model. *IEEE Signal Processing Letters*, 25(8):1181–1185, 2018.
- [13] M. Mauch and S. Ewert. The Audio Degradation Toolbox and its application to robustness evaluation. In *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR 2013)*, Curitiba, Brazil, 2013.
- [14] S. Meignier and T. Merlin. LIUM SpkDiarization: An open source toolkit for diarization. In *CMU SPUD Workshop*, Dallas, United States, 2010.
- [15] K. K. Rachuri, C. Mascolo, M. Musolesi, and P. J. Rentfrow. SociableSense: Exploring the trade-offs of adaptive sampling and computation offloading for social sensing. In *Proceedings of the 17th annual international conference on Mobile computing and networking - MobiCom '11*. ACM Press, 2011.
- [16] M. Trajanoska, M. Katrašnik, J. Lukan, M. Gjoreski, H. Gjoreski, and M. Luštrek. Context-aware stress detection in the aware framework. In M. Luštrek, R. Piltaver, and M. Gams, editors, *Proceedings of the 21st International Multiconference INFORMATION SOCIETY – IS 2018*, volume A, pages 25–28, 2018.
- [17] S. E. Tranter and D. A. Reynolds. An overview of automatic speaker diarization systems. *IEEE Transactions on audio, speech, and language processing*, 14(5):1557–1565, 2006.

Measurement of Bradykinesia for the Detection of Parkinson's Disease

Edi Čebokli
University of Ljubljana Faculty
of Computer and Information
Science
Večna pot 113, 1000
Ljubljana, Slovenia
ec9381@student.uni-lj.si

Dejan Georgiev
University Medical Centre
Division of neurology
Zaloška 7, 1000
Ljubljana, Slovenia
dejan.georgiev@kclj.si

Jure Žabkar
University of Ljubljana Faculty
of Computer and Information
Science
Večna pot 113, 1000
Ljubljana, Slovenia
jure.zabkar@fri.uni-lj.si

ABSTRACT

Non-motor and motor symptoms that are linked with Parkinson's disease are often clinically assessed by neurologists using the Unified Parkinson's Disease Rating Scale (UPDRS). UPDRS scores are described as qualitative and are dependent on neurologist's experience. Consequently, clinical scores may differ among neurologists. We develop an application for measuring bradykinesia in the UPDRS finger tapping task, with which patients are recorded with a depth camera and by analyzing videos, given a more objective rating. In the first stage, we detect touches and thumb's and pointer's fingertips. Following, we calculate distances between the fingertips. From distances we then extract finger tapping features. We record a group of people with Parkinson's disease and a control group. Furthermore, we define a model that best separates instances with different UPDRS scores. Considering the small number of training data, the model successfully separates the instances, however, we need to obtain more data for classification.

Keywords

bradykinesia, finger tapping, Unified Parkinson's Disease Rating Scale

1. INTRODUCTION

Parkinson's disease is a progressive neurodegenerative chronic disease, where early diagnosis is of utmost importance for inhibition of further progression of the disease and the onset of more serious symptoms that do not effect only quality of life [5]. Due to the lack of a diagnostic test, Parkinson's disease is often misdiagnosed or overlooked because of common symptoms with other diseases. Bradykinesia, which is one of the motor symptoms of Parkinson's disease, refers to the slowness of movement, decrease in amplitude and speed during performing repetitive movements of body segments [4].

For assessment neurologists usually use the Unified Parkinson's Disease Rating Scale (UPDRS) [6], that consists of four parts. The first part deals with self-evaluation of non-motor experiences of daily living, second with self-evaluation of motor experiences of daily living. The third part deals with motor examination and the fourth part assesses complications in treatment.

We develop an application for assessing finger tapping, which is one of the motor tasks examined in the UPDRS, with

a more objective rating. Finger tapping task consists of 10 taps, where the patient is instructed to tap their index and thumb fingers 10 times as quickly and as big as possible. In the first stage we detect touches and thumb's and pointer's fingertips, then calculate distances between the tips. From the distances, we define features that represent UPDRS characteristics such as amplitude decrements, slowness, interruptions, hesitations and halts. Individual tasks in UPDRS are rated with a score, ranging from 0 to 4. Both the left and the right hand is rated, separately. Requirements for each score of the finger tapping task are as follows:

- 0: Normal: No problems.
- 1: Slight: Any of the following: a) the regular rhythm is broken with one or two interruptions or hesitations of the tapping movement; b) slight slowing; c) the amplitude decrements near the end of the 10 taps.
- 2: Mild: Any of the following: a) 3 to 5 interruptions during tapping; b) mild slowing; c) the amplitude decrements midway in the 10-tap sequence.
- 3: Moderate: Any of the following: a) more than 5 interruptions during tapping or at least one longer arrest (freeze) in ongoing movement; b) moderate slowing; c) the amplitude decrements starting after the 1st tap.
- 4: Severe: Cannot or can only barely perform the task because of slowing, interruptions or decrements.

Many studies use accelerometers and gyroscopes attached to the fingertips [8, 9, 3, 2], however they may have an impact on the patient's performance of the finger tapping task. A depth camera, similar to ours, was used in [7] where multiple motor tasks were examined. 3Gear software development kit was used to detect fingertips and touches, for which a license is currently unobtainable.

2. APPLICATION IMPLEMENTATION

Our application includes a simple user interface, logic for recording and analyzing recorded videos, and a database for storing and reviewing results. Recording is done with an ASUS Xtion PRO LIVE depth camera. The resolution of the depth image is set at a maximum of 640×480 pixels



Figure 1: Example of a hand perpendicular to the camera's direction of recording.

at 30 frames per second. The following rules were set for recording for better results:

- There must be only one hand in the scene,
- the hand should be positioned as perpendicular as possible to the camera's direction of recording as visible in figure 1,
- the hand must be at least 0.6 meters away from the camera due to the range of the camera and must not be too far,
- other fingers should not be curled into a fist,
- recording should take place in a room not too exposed to sunlight,
- fingertips must be tapped 10 times as fast as possible and as big as possible.

2.1 Fingertip and Touch Detection

In the first stage, the hand needs to be separated from the background. Since patients are recorded in a way that the hand is the closest object, the closest pixels in a defined area can be taken. We also crop the image to exclude a portion of the arm we have no interest in.

Firstly, to detect fingertips we blur the image, using a median filter. Next, we remove excessive fingers using Canny edge detector. The hand is not always completely perpendicular to the direction of recording, so thresholds of Canny edge detector are not set too high, otherwise needed parts of the hand could be deleted. After deletion, we apply a morphological operation of closing to fill any holes that formed.

Occasionally excessive fingers do not get deleted with Canny edge detector. We use an algorithm that reduces an image to a 1-pixel wide skeleton. We iterate through the skeleton pixels and check for an adequate difference in depth of neighbouring pixels. Pixels further away get deleted along with the branch of the skeleton they were on. Two closest tips of branches are accepted as index and thumb fingertips, where the higher positioned tip is the index fingertip.

After deletion, we apply a morphological operation of dilation on the skeleton and then extract contours from the image. The largest contour represents the whole hand, in

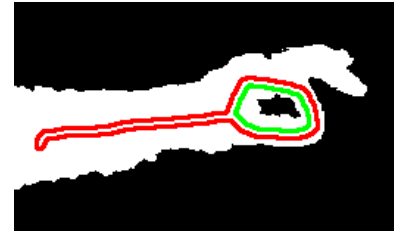


Figure 2: Example of obtained contours in case of a touch.

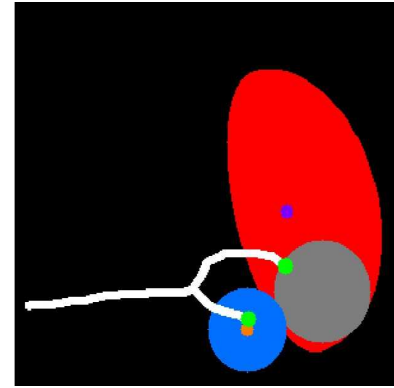


Figure 3: Typical regions of interest and the skeleton of a hand. Green circles represent the fingertips, the orange circle is the median of the fingertips of the thumb, and the purple circle is the median of the fingertips of the index fingers. The blue circle represents the first region, the gray region the second region, and the red image represents the third region composed of two parts.

case of a touch, a smaller contour on the inside of the fingers, as illustrated in figure 2, is formed, with which a touch is detected.

2.2 Correcting Misdetections

The closest points do not always represent the tips of the thumb and index fingers. To correct fingertip detection errors, we calculate the median of index and thumb fingertips every 20 images where a touch has not been detected. This way we reduce the impact of smaller hand movements and amplitude decrements. Based on the distance between the medians of fingertips, we define 3 regions of interest. The first one is a circle with its center in the median of the thumb's fingertips. The second one is located in the middle of the distance offset from the hand. The last one is composed of an angled ellipse and semi-ellipse, so it captures the index finger movement better. Regions are displayed in figure 3. Most of misdetections occur when there is an excessive branch originating near the joint of the index finger where its tip is closer than the index's fingertip. By making the 3rd region's ellipse narrower, we can avoid those misdetections. Tips of branches are accepted as fingertips based on if the tip is in the regions and the distance to the centres of regions. We choose the tip closest to the centre of the first region as the thumb fingertip and the tip closest to the centre of the second region as the index fingertip. Besides,

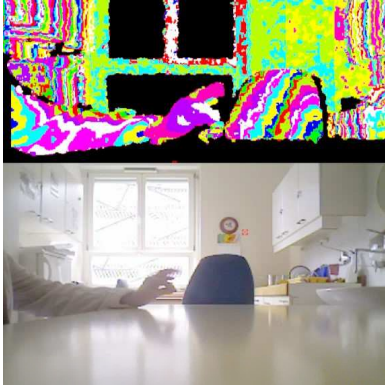


Figure 4: Errors in measuring depth. The depth image is represented with colored bands (top image).

we calculate the median length of index and thumb fingers. Branches where the tip is located on must be approximately the right length, for it to be considered a thumb or index finger. Next, we calculate z-score for every tip of a branch for x and y coordinates calculated as $z = (x - \bar{x})/\sigma$, where x is the coordinate, \bar{x} is the mean of the coordinates and σ is the standard deviation of the coordinates. If absolute z-scores of both coordinates of a tip are under 3 and are the lowest of all other tips' absolute z-scores, we accept the tip as a fingertip, otherwise we mark the image's fingertips as undetected. Touches are detected if more than one contour is found in an image. The occurrence of a second contour does not always mean a touch occurred, so correction is necessary. Based on the size and position of all second contours we set a minimal size requirement and restrict the location of the contour's centre.

2.3 UPDRS Score Calculation and Results

To calculate the UPDRS score, we recorded a group of people with Parkinson's disease at a neurological clinic at the University Medical Centre Ljubljana. Patients were instructed to tap their thumb and index finger 10 times as fast as possible and as big as possible, i.e. 10 cycles of opening and closing. Each patient was given a clinical evaluation of the left and the right hand by a neurologist. We recorded 16 people. With recordings of left and right hands we got a total of 32 recordings. Out of the 32 recordings, all touches and fingertips were correctly detected in 12 recordings. Due to the limited space at the clinic, we had to move the camera between recordings of left and right hands, which caused problems. Depth measurement was also adversely affected by sunlight. The biggest reason for misdetections was the poor adherence to rules described in the beginning of Sec. 2. In some recordings, hands were not placed sufficiently perpendicular to the direction of recording, causing problems in detecting touches or fingertips or both. In other recordings, patients curled the rest of their fingers into a fist. Some patients tapped with their finger pads instead of their fingertips. Tapping with pads can easily turn into tapping with your thumb pad and the joint. Since we rely on the contours for detecting touches, such tapping, low resolution, camera distance, and other influences can cause errors in measuring depth and make it harder to find a contour. An example of such errors is shown in Fig. 4.

The UPDRS score consists of the slowness of finger movement, amplitude decrements, number of interruptions, hesitations and halts. For the purpose of identifying the listed properties, we extracted the following attributes from distances between the fingertips:

- mean, standard deviation and slope of amplitudes (pAmp, sdAmp, sAmp),
- differences in percentage between the mean of all amplitudes except the last and the last amplitude, between the mean of the first and second half of amplitudes, and between the first and the mean of all other amplitudes (Damp1, Damp2, Damp3),
- mean, standard deviation, and slope of the mean opening, closing, and full-cycle velocities (mVopen, sdVopen, sVopen, mpVclose, sdVclose, sVclose, mVcyc, sdVcyc, sVcyc),
- mean, standard deviation and slope of maximum opening and closing velocities (mMVopen, sdMVopen, sMVopen, mMVclose, sdMVclose, sMVclose),
- mean, standard deviation and slope of percentage of time of a cycle to maximum opening velocity (mTMVopen, sdTMVopen, sTMVopen),
- mean, standard deviation and slope of cycle duration (mLen, sdLen, sLen),
- mean, standard deviation and slope of touch duration (mTou, sdTou, sTou),
- mean, standard deviation and slope of mean acceleration (mA, sdA, sA),
- number of interrupts (Inter),
- hesitation index (Hesit),
- number of halts (Halt).

Amplitudes were calculated as the maximum distance between fingertips in a cycle. We calculate decrements as described in the UPDRS with Damp1, Damp2, Damp3 and by using linear regression. We divide speed into mean opening, closing velocity and the velocity of the entire cycle. Opening velocity is defined by absolute differences of distances from the beginning of opening to the amplitude in the cycle divided by the travel time, similarly, closing velocity with absolute differences of distances from the amplitude to the end of closing, and velocity of the entire cycle from the beginning of opening to the end of closing. In addition to the average velocity, speed is also expressed with means of cycle and touch duration, maximum opening and closing velocities, and average accelerations in cycles. We obtain changes with linear regression. Halts are longer arrests in movement. We detect halts if the difference between the cycle duration and the approximation, calculated by Theil-Sen estimator, is above a certain threshold as calculated in [7]. Hesitations occur at the start of an opening or closing. We detect hesitations with an index, calculated by summing up the absolute differences between the percentage of time of a cycle to the maximum opening velocity and the approximation, calculated by Theil-Sen estimator, similarly as done in [8]. In a

CmoPy: Constrained Multiobjective Optimization in Python

Aljoša Vodopija
Jožef Stefan Institute and
Jožef Stefan International Postgraduate School
Jamova cesta 39
SI-1000 Ljubljana, Slovenia
aljosa.vodopija@ijs.si

Bogdan Filipič
Jožef Stefan Institute and
Jožef Stefan International Postgraduate School
Jamova cesta 39
SI-1000 Ljubljana, Slovenia
bogdan.filipic@ijs.si

ABSTRACT

Python is one of the most frequently used programming languages for solving multiobjective optimization problems (MOPs). Although there exist Python packages covering a wide range of multiobjective optimization tools, there is still a lack of implemented constraint handling techniques (CHTs). This paper introduces a new package for Constrained Multiobjective Optimization in Python (*CmoPy*). It describes the package implementation, the included CHTs and benchmark problems, and shows examples of using the package. With five state-of-the-art CHTs and 22 benchmark constrained MOPs, *CmoPy* is currently the most comprehensive Python package for constrained multiobjective optimization.

Keywords

Python, constrained multiobjective optimization, multiobjective evolutionary algorithms, constraint handling techniques, NSGA-II algorithm

1. INTRODUCTION

Real-world optimization problems regularly involve both multiple objectives and constraints. Such problems are called *constrained multiobjective optimization problems* (CMOPs). A CMOP can be formulated as

$$\begin{aligned} & \text{minimize} && f_m(x), \quad m = 1, \dots, M \\ & \text{subject to} && g_n(x) \leq 0, \quad n = 1, \dots, N \end{aligned}$$

where $x = (x_1, \dots, x_D)^T$ is a *decision vector* (solution), $f_m : S \rightarrow \mathbb{R}$ are *objective functions*, $g_n : S \rightarrow \mathbb{R}$ *constraint functions*, $S \subseteq \mathbb{R}^D$ is a *decision space* of dimension D , and M and N are the numbers of objectives and constraints, respectively. Additionally, $f_m(x)$ is an *objective value*, $g_n(x)$ a *constraint value*, $\phi_n(x) = \max(g_n(x), 0)$ *constraint violation* and $\phi(x) = \sum_n \phi_n(x)$ the *overall constraint violation*. A solution satisfying all the constraints is a *feasible solution* and otherwise an *infeasible solution*.

A feasible solution $x^{(1)}$ is said to *dominate* a feasible solution $x^{(2)}$ if and only if $f_m(x^{(1)}) \leq f_m(x^{(2)})$ for all $m \in \{1, \dots, M\}$ and $f_m(x^{(1)}) < f_m(x^{(2)})$ for at least one index m . A feasible solution x^* is a *Pareto-optimal solution*, if there are no feasible solutions from S dominating x^* . The set of all Pareto-optimal solutions from S is called a *Pareto-optimal set* and its image in the objective space a *Pareto front*. The goal of multiobjective optimization is to find an approximation of the Pareto front that represents trade-offs between the objectives.

Python is one of the most frequently used programming languages for solving multiobjective optimization problems. Due to its simplicity, versatility and accessibility of numerous open source optimization tools, it is suitable for academic research as well as for real-world applications. In the Python Package Index—the biggest repository of Python software—there are currently six packages dealing with (nonlinear) multiobjective optimization problems: *DEAP* [7], *inspyred* [9], *jMetalPy* [1], *PaGMO 2.0* [2], *Platypus* [10] and *pymoo* [3]. All these packages implement the most common evolutionary algorithms for multiobjective optimization, such as Nondominated Sorting Genetic Algorithm II (NSGA-II) [5], Multiobjective Optimization Evolutionary Algorithm Based on Decomposition (MOEA/D) [18] and others.

However, constraint handling techniques (CHTs) are not explicitly addressed in these packages. In *jMetalPy*, *PaGMO 2.0* and *Platypus*, only the constrained-domination principle (CDP) [5] is implemented to deal with constraints. In *pymoo* and *inspyred*, constraints are not considered at all, while *DEAP* implements a CHT based on a penalty function. For this reason, there is still a great need for a more comprehensive package covering the area of constrained multiobjective optimization.

In this paper, we present a new package for constrained multiobjective optimization named *CmoPy*, which stands for Constrained Multiobjective Optimization in Python. The package is designed as a potential functionality in the *SciPy* Python tool [12]. It implements five state-of-the-art CHTs and an ensemble method capable of including multiple CHTs to handle constraints. In addition, several widely used CMOPs are included for benchmarking purposes.

The rest of this paper is organized as follows. Section 2 describes the implementation of *CmoPy*. Section 3 is dedicated to the CHTs implemented in the package, while the included CMOPs are covered in Section 4. In Section 5, some examples of using the package are shown. Finally, Section 6 summarizes the *CmoPy* presentation and provides ideas for future work.

2. PACKAGE IMPLEMENTATION

The main function in *CmoPy* is `nsga_ii`. When called, this function executes the NSGA-II algorithm to solve the given CMOP. The `nsga_ii` function resembles the original NSGA-II [5] in all segments except for minor modifications in the survivor selection phase, where the population is selected for

the new generation, to allow for the inclusion of additional CHTs (see [17] for details on these modifications and Section 3 for a summary of CHTs implemented in *CmoPy*). The input and output parameters of the `nsga_ii` function are described below.

The input parameters of the `nsga_ii` function:

- **problem (Problem)**: A custom object including four parameters:
 - **fun (callable)**: A Python callable (function) consisting of objective and constraint functions. Must be of the form $f(x, *args)$ where x is the argument in the form of a 1-D `ndarray` and `args` is a tuple of all additional fixed parameters needed to fully specify the objective and/or constraint functions. The output must be a 1-D `ndarray` of objective and constraint values. Note that only inequality constraints of the form $g(x) \leq 0$ are accepted. Equality and other forms of constraints need to be reformulated as inequality constraints.
 - **bounds (list of tuple)**: Bounds of the form (min, max) that define the lower and upper bounds for the arguments of `fun`.
 - **no_cons (int, optional)**: The number of constraints. Default is 0.
 - **args (tuple, optional)**: Any additional parameters needed to completely specify the objective and/or constraint functions. Default is `None`.
- **max_iter (int, optional)**: The maximum number of generations to be executed by the optimizer. Default is 250.
- **pop_size (int, optional)**: Number of solutions in the population. Default is 100.
- **max_fun (int, optional)**: The maximum number of function evaluations. Default is 25,000.
- **mut_prob (float, optional)**: The probability for a solution to be mutated. Default is $1/D$.
- **cross_prob (float, optional)**: The probability for parents to be altered by crossover. Default is 0.9.
- **mut_eta (int, optional)**: The distribution index for the polynomial mutation. Default is 20.
- **cross_eta (int, optional)**: The distribution index for the simulated binary crossover. Default is 20.
- **seed (int, optional)**: This parameter controls the seeds of the stochastic processes applied during the algorithm run. If no value is specified, a random seed is used.
- **init (str or ndarray, optional)**: The type of the population initialization. It can be "lhs" for the latin hypercube sampling, "rand" for random population initialization, or a `ndarray` of predefined solutions. In the last case, the parameter `pop_size` is equal to the number of rows in the array. Default is "lhs".

- **cht (tuple of str, optional)**: The required constraint handling technique. It can be "nds" for non-dominated sorting, "cdp" for constrained-domination principle, "dpf" for dynamic penalty function, "str" for stochastic ranking, "mcr" for multiple constraint ranking, or a tuple of any set of these methods. In the latter case, the ensemble of specified CHTs is used. The default value is "nds" if there are no constraints and "cdp" otherwise.

The output parameter is a custom object named `Result`. It is a multiobjective extension of the object `OptimizeResult` used to represent results in the optimization module of *SciPy* [12]. The object `Result` includes the following parameters:

- **x (ndarray)**: Solutions from the final population.
- **x_all (ndarray)**: All nondominated feasible solutions found during the entire optimization run.
- **success (bool)**: Whether or not the optimizer exited successfully.
- **status (int)**: The optimizer termination status.
- **message (str)**: The description of the termination cause.
- **fun (ndarray)**: The objective and constraint values of solutions from the final population.
- **fun_all (ndarray)**: The objective and constraint values of all nondominated feasible solutions found during the entire run.
- **nfev (int)**: The number of the `fun` function evaluations.
- **nit (int)**: The number of generations executed by the optimizer.
- **maxcv (float)**: The maximum overall constraint violation.

The implementation of *CmoPy* follows the guidelines for contributing to *SciPy*. The only dependency apart from *SciPy* needed to use the package is *NumPy* [14].

3. CONSTRAINT HANDLING TECHNIQUES

In *CmoPy*, there are five widely used CHTs and an ensemble method combining any desired set of single techniques:

- **Nondominated sorting [5]**: This method selects the new generation of solutions according to the dominance relation, not considering constraint violations at all. This method is used as the default CHT for unconstrained problems.
- **Constrained-domination principle [5]**: This CHT can be seen as an extension of the nondominated sorting, where feasible solutions dominate infeasible ones, and infeasible solutions are ranked according to the overall constraint violation. This method is used as the default CHT for constrained problems.

- Dynamic penalty function [6]: This method augments the fitness of a solution by adding a penalty that is proportional to the overall constraint violation. The penalty pressure is increased in each generation.
- Stochastic ranking [15]: This CHT uses a bubble-sort-like process to rank solutions in the population. Two feasible solutions are always compared based on their fitness. On the other hand, if at least one of the solutions is infeasible, then a random decision is made on whether the two solutions are compared based on their fitness or constraint violation.
- Multiple constraint ranking [8]: In this approach, the solutions are ranked based on their fitness and constraint violation. If there are no feasible solutions, only the rank generated from constraint violation is considered, otherwise a combination of both ranks is taken into account.
- Ensemble of CHTs [17]: This approach combines multiple single CHTs into an ensemble-based ranking. The solutions are ranked based on a quality measure which is averaged over all techniques in the ensemble.

4. INCLUDED BENCHMARK PROBLEMS

The *CmoPy* package contains 22 CMOPs that are frequently used for benchmarking purposes (see, for example, [4, 16]). Table 1 summarizes these problems considering three basic characteristics: the dimension of the decision space, the number of objectives and the number of constraints.

Table 1: Characteristics of the CMOPs included in *CmoPy*: dimension of the decision space D , number of objectives M and number of constraints N .

CMOP	D	M	N
Belegundu [4]	2	2	2
Binh 1 [4]	2	2	2
Binh 2 [4]	2	3	2
C1-DTLZ1 [11]	$M + 4$	≥ 2	1
C1-DTLZ3 [11]	$M + 9$	≥ 2	1
C2-DTLZ2 [11]	$M + 9$	≥ 2	1
C3-DTLZ1 [11]	$M + 4$	≥ 2	M
C3-DTLZ4 [11]	$M + 4$	≥ 2	M
Car-side impact [11]	3	10	7
DTLZ8 [4]	30	3	3
DTLZ9 [4]	30	3	2
Jimenez [4]	2	2	4
Kita [4]	2	2	3
Obayashi [4]	2	2	1
Osyczka 1 [4]	2	2	2
Osyczka 2 [4]	6	2	6
Srinivas [4]	2	2	2
Tamaki [4]	3	3	1
Tanaka [4]	2	2	2
Vibrating platform [13]	2	5	5
Viennet [4]	2	3	2
Water resource planning [11]	5	7	3

5. EXAMPLES OF USING THE PACKAGE

This section illustrates the use of *CmoPy* on the well-known Srinivas [4] problem. It also shows an example of adding a new CMOP to the package.

We first need to install *CmoPy*. This can be achieved by running the following commands:

```
$ git clone https://gitlab.com/cmopy/cmopy.git
$ cd cmopy
$ python setup.py
```

In the following example, the NSGA-II algorithm is run to solve the Srinivas problem. The population size is set to 80 solutions (`pop_size=80`), while the number of generations is kept at the default value of 250. In addition, CDP is used for handling constraints as a default option.

```
>>> from cmopy.optimize import nsga_ii
>>> from cmopy.problems import srinivas, Problem
>>> nsga_ii(srinivas, pop_size=80)
```

To add new CMOPs to *CmoPy*, we need to implement the `problem` object introduced in Section 2. In addition to the callable object (`test_fun`) consisting of the objective and constraint functions, we need to specify the bounds (`bounds`) and the number of constraints (`no_cons`). The following example shows the implementation of the Tanaka problem [4].

```
>>> import numpy as np
>>> def test_fun(x):
...     x1, x2 = x
...     g1 = 1 - x1 ** 2 - x2 ** 2 + \
...         0.1 * np.cos(16 * np.arctan(x1 / x2))
...     g2 = (x1 - 0.5) ** 2 + \
...         (x2 - 0.5) ** 2 - 0.5
...     return np.array([x1, x2, g1, g2])
>>> bounds = [(0, np.pi)] * 2
>>> no_cons = 2
>>> test_cmap = Problem(test_fun, bounds, no_cons)
```

At this point, we can run the NSGA-II algorithm to solve the implemented problem. Here, we use the dynamic penalty function to handle constraints (`cht="dpf"`).

```
>>> nsga_ii(test_cmap, cht="dpf")
```

If we want to run an ensemble combining multiple CHTs, we need to specify all the desired techniques in a tuple. The `nsga_ii` function automatically detects that an ensemble method is required. In the following example, the optimizer uses an ensemble combining the constrained-domination principle and the dynamic penalty function to handle constraints (`cht=("cdp", "dpf")`).

```
>>> nsga_ii(test_cmap, cht=("cdp", "dpf"))
```

Figure 1 shows the Pareto front approximations for the Srinivas problem (left) and the Tanaka problem (right) obtained after running the above commands.

6. CONCLUSIONS

We introduced *CmoPy*, a Python package designed for constrained multiobjective optimization. Unlike other Python packages for multiobjective optimization, it contains a comprehensive set of CHTs. While other packages usually implement one simple method to deal with constraints, in *CmoPy* there are five state-of-the-art CHTs and an ensemble-based

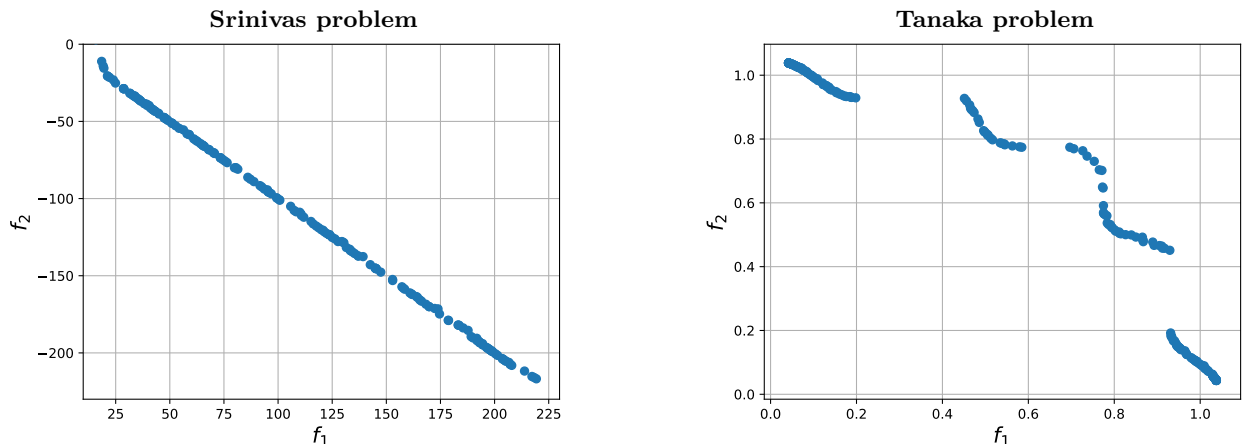


Figure 1: Pareto front approximations generated by *CmoPy* for the Srinivas problem (left) and the Tanaka problem (right). The former was solved by handling constraints with CDP, while the latter by applying an ensemble of CHTs.

method combining multiple single techniques. Moreover, an extensive set of benchmark CMOPs with various characteristics is also included in *CmoPy*.

In the future, we plan to extend the package functionality by adding other multiobjective optimizers, for example, MOEA/D, and integrating the methods for visualizing the results and algorithm performance.

7. ACKNOWLEDGMENTS

The authors acknowledge the financial support from the Slovenian Research Agency (Young researcher program and Research core funding No. P2-0209).

8. REFERENCES

- [1] A. Benítez-Hidalgo, A. J. Nebro, J. García-Nieto, I. Oregi, and J. D. Ser. jMetalPy: A Python framework for multi-objective optimization with metaheuristics. <http://arxiv.org/abs/1903.02915>, 2019.
- [2] F. Biscani et al. PaGMO 2.0. <https://github.com/esa/pagmo2>, 2019.
- [3] J. Blank and K. Deb. pymoo: Multi-objective optimization in Python. <https://pymoo.org>, 2019.
- [4] C. A. Coello Coello, G. B. Lamont, and D. A. Van Veldhuizen. *Evolutionary Algorithms for Solving Multi-Objective Problems*, chapter 4, pages 179–238. Springer-Verlag Berlin Heidelberg, 2nd edition, 2007.
- [5] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
- [6] A. E. Eiben and J. E. Smith. *Introduction to Evolutionary Computing*, chapter 8, pages 129–151. Natural Computing Series. Springer-Verlag Berlin Heidelberg, 2003.
- [7] F.-A. Fortin, F.-M. De Rainville, M.-A. Gardner, M. Parizéau, and C. Gagné. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, 13(1):2171–2175, 2012.
- [8] R. D. P. Garcia, B. S. L. P. de Lima, A. C. D. C. Lemonge, and B. P. Jacob. A rank-based constraint handling technique for engineering design optimization problems solved by genetic algorithms. *Computers & Structures*, 187:77–87, 2017.
- [9] A. Garrett. inspyred: Bio-inspired algorithms in Python. <https://github.com/aarongarrett/inspyred>, 2019.
- [10] D. Hadka. Platypus: Multiobjective optimization in Python. <https://github.com/Project-Platypus/Platypus>, 2019.
- [11] H. Jain and K. Deb. An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, Part II: Handling constraints and extending to an adaptive approach. *IEEE Transactions on Evolutionary Computation*, 18(4):602–622, 2014.
- [12] E. Jones, T. Oliphant, and P. Peterson. SciPy: Open source scientific tools for Python. <http://www.scipy.org>, 2019.
- [13] A. Messac. Physical programming: Effective optimization for computational design. *American Institute of Aeronautics and Astronautics Journal*, 34(1):149–158, 1996.
- [14] T. E. Oliphant. *A guide to NumPy*, volume 1. Trelgol Publishing USA, 2006.
- [15] T. P. Runarsson and X. Yao. Stochastic ranking for constrained evolutionary optimization. *IEEE Transactions on Evolutionary Computation*, 4(3):284–294, 2000.
- [16] R. Tanabe and A. Oyama. A note on constrained multi-objective optimization benchmark problems. In *2017 IEEE Congress on Evolutionary Computation, CEC 2017*, pages 1127–1134. IEEE, 2017.
- [17] A. Vodopija, A. Oyama, and B. Filipič. Ensemble-based constraint handling in multiobjective optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion, GECCO '19*, pages 2072–2075. ACM, 2019.
- [18] Q. Zhang and H. Li. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6):712–731, 2007.

Designing an Intelligent Cognitive Assistant for Behavior Change in Mental Health

Tine Kolenik

Jožef Stefan Institute & Jožef Stefan
International Postgraduate School
Jamova cesta 39
1000 Ljubljana, Slovenia
+386 1 477 3807
tine.kolenik@ijs.si

Martin Gjoreski

Jožef Stefan Institute & Jožef Stefan
International Postgraduate School
Jamova cesta 39
1000 Ljubljana, Slovenia
+386 1 477 3812
martin.gjoreski@ijs.si

Matjaž Gams

Jožef Stefan Institute
Jamova cesta 39
1000 Ljubljana, Slovenia
+386 1 477 3644
matjaz.gams@ijs.si

ABSTRACT

The paper describes intelligent cognitive assistant technologies and how they can be used efficiently for behavior change in mental health, namely stress, anxiety and depression. It then reviews the state of the art of such cases, focusing on three different assistants and their experimental results. It raises some critical issues with the state of the art and the field itself, namely the lack of standardized evaluation and that current technologies do not take advantage of artificial intelligence and behavior sciences advances. The paper proposes its own comprehensive cognitive architecture for such an assistant, relying on the integration of behavior change theories and cohesive user modeling (together referred to as the theory of mind) into a cognitive architecture. In addition to linguistic input, the architecture includes biophysiological input for affect recognition purposes. Reinforcement learning and the use of the principle of multiple knowledge are proposed as the main drivers for strategy adaptation in relation to helping users. The paper attempts to fill the gaps in the works related to this field, which are mostly closed source, believing that an overview of the field, its issues and a proposed design will enrich the current academic landscape.

Keywords

Behavior change, cognitive architecture, intelligent cognitive assistant, mental health, user modeling and profiling.

1. INTRODUCTION

Intelligent cognitive assistants (ICAs) have been described as the next revolution in human-computer coexistence, particularly because of the idea of conversing with people in natural language [1]. The technology has relatively ancient roots in the history of artificial intelligence (AI), consisting of famous examples such as Weizenbaum's simulation of a Rogerian psychotherapist called ELIZA [2]. However, the technology has only recently laid the foundations for broad adoption in the form of ICAs such as Alexa and Siri as well as more domain-specific agents, and has been thus on the rise in terms of financing and research [1]. ICAs, which can be deployed as virtual agents or robots, are being made to: understand context; be adaptive and flexible; learn and develop; be autonomous; be communicative, collaborative and social; be interactive and personalized; be anticipatory and predictive; perceive; act; have internal goals and motivation; interpret; and reason. To achieve this in ICAs, they are embedded with a cognitive architecture (CogA), a "hypothesis about the fixed structures that provide a mind, whether in natural or artificial systems, and how they work together – in conjunction with knowledge and skills embodied within the architecture – to yield intelligent behavior in a diversity of complex environments" [3].

ICAs have recently shown great potential as persuasive technology in the domain of behavior change (BC). BC is a phenomenon that is considered to be a temporary or permanent effect on an individual in terms of their behavior, attitude and other mental states as compared to their past [4]. Persuasive technology can be defined as technology designed for attempting to "change attitudes or behaviors or both (without using coercion or deception)" [4, p. 20]. Persuasive technologies are already heavily used in the areas of health and wellness, where AI tracks people's behavior as well as physiological and mental states to motivate them to make better dietary decisions and exercise more along with offering people psychotherapeutic help in natural language [5]. To achieve this, user modelling and profiling (UMP) is extremely important in order to understand users' intentions, needs and states in relation to their psychographics, which can then be used for personalized and thus more effective BC through carefully selected outputs [6].

Of particular importance for persuasive technologies is the area of psychotherapeutic help. Stress, anxiety and depression (SAD) are prevalent and rising problems, with research revealing that more and more people are suffering from at least one of these mental issues, with figures in some groups reaching 71% for stress, 12 % for anxiety disorder and 48% for depression [7]. Slovenia, already struggling with the second highest suicide rate in Europe [8], also suffers from a serious shortage of officially recognized and certified mental health professionals as well as from a lack of regulation [9]. This opens the door to ICAs, as they are not only generally free to use (making help available to socioeconomically disadvantaged people) and available 24/7 (so people do not have to wait for their next therapy session), people tend to be more comfortable disclosing their feelings to an ICA than to a person [10], ICAs are available in remote locations, ICAs reduce burden on the healthcare system and its practitioners, and overall reduce barriers to mental healthcare access [11]. Such ICAs do not serve to replace professionals, but to complement them.

This introduction is followed by an overview of the state of the art. Afterwards, our design process and related work of constructing a psychotherapeutic ICA (PICA) are presented. Our work is in the phases of planning, analysis and design of the Systems development life cycle [12]. The paper focuses mostly on the computational (what the system does and why) and algorithmic level (how the system does what it does) of Marr's 3-level design hierarchy for AI systems [13]. Since papers on designs and their hurdles for PICAs are almost non-existent (as opposed to other kinds of ICAs), this work should find a useful place in the growing ICA research bibliography.

2. STATE-OF-THE-ART OVERVIEW

The chapter focuses on ICAs for BC in psychotherapy and mental health, namely for SAD. Various mechanisms of these systems are highlighted for the purposes of this work.

PICAs seem unique among ICAs, especially compared to Q&A ICAs. Users reveal personal information more freely, which makes PICAs more successful in their goals [14]. PICAs and their respective users also form a more longitudinal relationship. The interactions are not a one-off, where it is difficult to understand the users and act immediately with efficient strategies. This makes PICAs able to learn from historical interactions and improve in offering personalized psychotherapeutic help. However, since the use of ICAs as persuasive technology for BC is a recent endeavor, the pool of existing PICAs is small. The selection process of PICAs for this overview was based on the condition that they had been researched in an ecological environment (interacting with end users) with empirical experiments (e.g., randomized controlled trial). The metric is therefore being successful in relieving symptoms of SAD (a very indirect metric for determining which PICA performs best and why). PICAs cannot be compared in the same way as Q&A ICAs, where there are answers that need to be mapped to some collection of questions. PICAs must therefore be measured by their direct impact on the users [15], which makes them much more personalized in their acting, causing evaluation to be subjective and indirect. Our last selection metric was for PICAs to be text-based.

One of the most recent successful PICAs is Woebot [14]. Its overarching methodology is a “decision tree with suggested responses that accepts natural language inputs with discrete sections of natural language processing techniques embedded at specific points in the tree to determine routing to subsequent conversational nodes” [14, p. 3]. It gathers data on users’ moods, goals, expectations and similar to build a user model and dispatch an intervention in the form of educational content, personalized messages, contextual strategies and scripted advice. In a randomized controlled trial, Woebot delivered better treatments to people than the government-approved self-help material, where the PICA relieved depression symptoms by app. 20% on average (Figure 1).

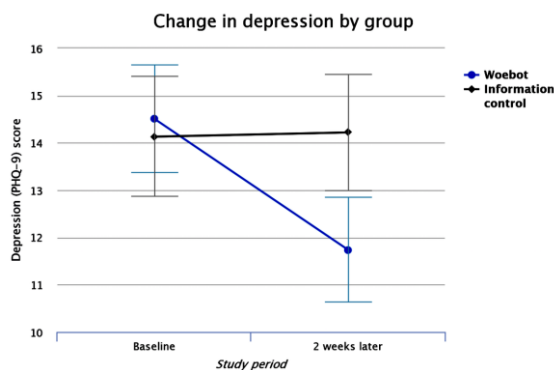


Figure 1 [14, p. 6]: Change in depression, control (‘Information control’) vs. test (‘Woebot’) group.

Another PICA is by Yorita, Egerton, Oakman, Chan and Kubota [16]. The PICA is built on the Belief-Desire-Intention (BDI) CogA (Figure 2). It comprises of three core models: “a conversation model for acquiring state information about the individual, measuring their stress level, a Sense of Coherence (SOC) model for evaluating the individuals state of stress, and Peer Support model, which uses the SOC to select a suitable peer

support type and action it” [16, p. 3762], meaning it selects a suitable strategy based on the user model. ‘Stress level’ refers to scores from stress questionnaires, while ‘state of stress’ refers to three dimensions related stress that users actively try to improve: comprehensibility, manageability, meaningfulness.

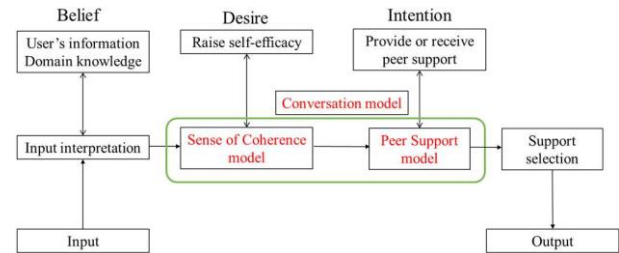


Figure 2 [16, p. 3762]: BDI CogA.

The experiment in the research found that the PICA had constructed more and more accurate models of its users, who managed stress better with each day of usage.

A PICA called Tess “reduce[s] self-identified symptoms of depression and anxiety” [17, p. 1]. It is crucially based on a wide emotion ontology, which it uses to determine and match users’ emotions from their inputs. It uses stateful conversational models with scripted conversations, consisting of a natural language understanding module, dialogue state manager and natural language response generator. Improving strategies include asking for feedback as well as using user journaling data Tess gathers. Using Tess for psychotherapeutic interventions for mental health research showed that it significantly reduces anxiety and depression symptoms – roughly 16% for group 1 and 15% for group 2 (Figure 3).

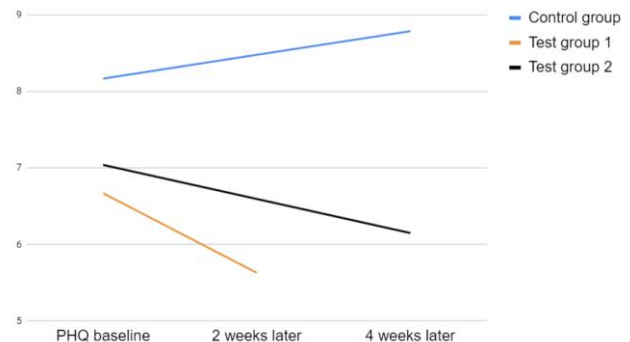


Figure 3 [17, p. 6]: Change in depression level (y-axis). Test group 1 used Tess for 2 weeks; 2 used it for 4 weeks. Control group used the government-suggested eBook on depression.

The PICAs described represent fully-formed agents with many mechanisms combined in a complete architecture. Most happen to be proprietary (closed source) and not available for technical scrutiny; there are no research papers that would describe them in detail, which holds true for most PICAs available online. Examining their architectures and implementation is therefore difficult, as noted by other researchers as well [18]. The overviewed work points at the PICAs being extremely goal-oriented and one-dimensional due to a lack of a contemporary UMP-based approach, a more coherent or theory-based CogA as well as a lack of inclusion if BC theories [5]. What follows is a description of a possible design and its hurdles towards a more effective PICA based on a coherent CogA, comprehensible UMP module and embedded BC domain knowledge.

3. THE PROPOSED DESIGN

This chapter describes the processes, related to the design of our PICA. First, an overview of contemporary methods of various fields used in ICAs is presented. Then, the necessity of embedded BC knowledge in relation to a comprehensive user model as well as a cohesive CogA for an effective PICA is argued through presenting the design and future implementation ideas.

For natural language understanding, dialogue-based applications are relevant for ICAs, where state-of-the-art applications use algorithms such as word2vec, Latent Semantic Analysis and deep learning methods [19]. Regarding UMP, there is a fairly accepted set of user characteristics that generally make artificial systems perform better when modelled: knowledge, beliefs, background, interests, preferences, goals, plans, tasks, needs, demographic information, emotional state, and context [20]. For CogAs, functional (as opposed to structural) architectures like BDI (and its various modifications) seem to be recommended [20].

Endowing a PICA with expert knowledge on BC has to be based on behavioral sciences advances in regards to human decision-making and similar phenomena [21], which has been recently combined with digital technologies, AI and big data. Many societal efforts are being put into creating persuasive technologies that would help, motivate, guide and affect people into bettering themselves and the world around them. One of the most powerful and effective contemporary BC concepts is the ‘nudge theory’¹. Nudge is “any aspect of the choice architecture that alters people’s behavior in a predictable way without forbidding any options or significantly changing their economic incentive” [21, p. 6]. We believe that incorporating nudges and similar BC techniques into our PICA is essential for its effective behavior.

To effectively dispatch nudges and other strategies, our PICA (Figure 4) will build and continually update a user model. It is not clear which data on users is the most optimal for mental health BC, but we intend to implement a dialogically delivered questionnaire [22] on the Big Five personality traits (B5) as a fundamental element on the longitudinal (or global) user aspect [23]. This will largely inform PICA’s BC strategies as B5 is an extremely successful psychological construct in determining what kind of influence is effective on a person as well as many other inferences [24]. In the less longitudinal aspects, determining SAD will be fundamental. This will again be achieved through dialogically delivered Depression Anxiety Stress Scales 21 questionnaire [25] to determine specific mental health symptoms. SAD scores will be refined on a regular basis, as reinforcement learning will be used to adapt PICA’s behavior according to the changes in the SAD model [26]. The re-strategizing timeframe will have to be determined experimentally for optimal strategy adaptation. Continuous sentiment analysis (relying on Slovene sentiment lexicon JOB 1.0 if in Slovene [27]) will be used for a relatively short-term (albeit more inaccurate) inference on the user’s emotional state [28]. Other user dimensions will comprise of as much of the previously listed accepted characteristics as is possible, feasible and sensible. Adding biophysiological measurements (e.g., heart rate, sweating rate and skin temperature) from smart bands is being considered as well. This may be used for automatic monitoring of SAD [29–32]. For example, at the beginning, the SAD answers can be used to label the biophysiological measurements. Once enough data is labeled, the labeled biophysiological measurements can be used to learn

personalized machine learning (ML) models for predicting SAD. The SAD predictions can be later utilized by the SAD model. This eases the user’s burden since once the ML models for predicting the SAD are learned, the user will answer SAD questionnaires less frequently. The learning and the management of the personalized ML models will be handled by the Affect recognition module. Our PICA’s strategy selection will be based on the principle of multiple knowledge [33] – which says that when high quality different viewpoints are sensibly filtered, adapted and combined, the result will be superior to the individual methods – and the predictive coding theory of cognition (PCT) [34]. There will be multiple competing strategies that will contend, adapt and enact according to multiple criteria from the interpreted linguistic input, user model and BC domain knowledge. A strategy (one response or a more long-term strategy) will be selected when a certain confidence threshold will be reached. If the threshold is not reached, necessary information on the user will be dialogically extracted until the probability that a certain strategy will be effective is reached. This approximately mimics PCT, but a strict formalization is needed in the future and further elaboration is currently out of scope for this paper.

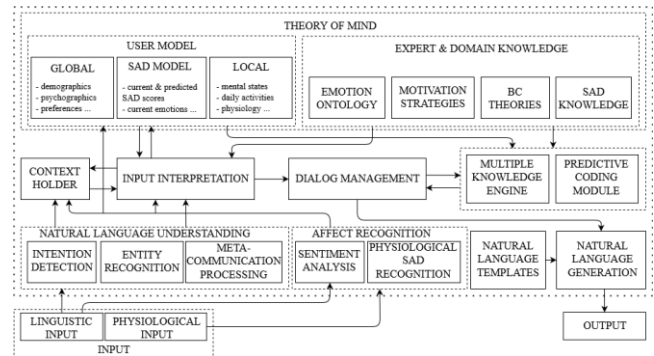


Figure 4: A tentative CogA for the described PICA.

4. CONCLUSIONS AND FUTURE WORK

This paper serves primarily the purpose of outlining the steps of planning, analysis and design of our PICA. This serves a variety of purposes: to highlight numerous problems in researching PICAs (the problems of existing PICAs being mostly proprietary and closed source, which means that there is a lack of research on PICAs that would illuminate technical details; a lack of standardized evaluation measures for PICAs), to contribute to a flourishing field of PICAs with a design paper, and to explicate our own unique ideas into a more coherent framework.

Our future work will require to define clear evaluation measures for PICAs. The lack of the latter points to a greater lack of a comprehensive overview on which CogAs are best suited for specific domains. This is all the more evident as there is little to no research on what user (or expert) data is needed for efficient PICAs. Our future work will continue with the Systems development life cycle steps (especially implementation; some work has already begun [35]) as well as with the third level of Marr’s 3-level design hierarchy – physical (the realization of the first two levels) – after more thorough completion of the analysis and design steps pending feedback of the current work when published. The implementation will go hand in hand with experiments, which will inform the design of the CogA. The experiment by Fitzgerald et al [14] (see Woebot in chapter 2) will be replicated for comparison purposes.

¹ Richard Thaler, its author, received the Nobel Prize for it.

We believe that the highlighted research areas and technologies show promise and potential in both effectiveness (use case) as well as openness to new explorations and progress, which makes our research not only sensible, but also necessary.

5. ACKNOWLEDGMENTS

This work is supported by Slovenian Research Agency's Young researchers postgraduate research funding.

6. REFERENCES

- [1] Oakley, J. 2018. *Intelligent Cognitive Assistants (ICA)*. Workshop Summary. IBM Almaden Research Center.
- [2] Weizenbaum, J. 1966. ELIZA—a computer program for the study of natural language communication between man and machine. *Commun. ACM*. 9, 1 (1966), 36-45.
- [3] Cognitive architecture. 2019. *Wikipedia*.
- [4] Fogg, B. J. 2002. *Persuasive technology*. Morgan Kaufmann, Burlington, MA.
- [5] Orji, R. and Moffatt, K. 2018. Persuasive technology for health and wellness: State-of-the-art and emerging trends. *Health Inform. J.* 24, 1 (2018), 66-91.
- [6] Hardcastle, S. J. and Hagger, M. S. 2016. Psychographic Profiling for Effective Health Behavior Change Interventions. *Front. Psychol.* 6, 1988 (2016).
- [7] Twenge, J. M. 2014. Time Period and Birth Cohort Differences in Depressive Symptoms in the U.S., 1982–2013. *Soc. Indic. Res.* 121, 2 (2014), 437-454.
- [8] Just over 56 000 persons in the EU committed suicide. 2018. *Eurostat*.
- [9] Kuralt, Š. 2015. Slovenija je po številu kliničnih psihologov na evropskem dnu. *Delo*.
- [10] Lucas, G. M., Gratch, J., King, A., and Morency, L. P. 2014. It's only a computer: virtual humans increase willingness to disclose. *Comp. Human. Behav.* 37 (2014), 94-100.
- [11] Mohr, D. C., Burns, M. N., Schueller, S. M., Clarke, G., and Klinkman, M. 2013. Behavioral Intervention Technologies: Evidence review and recommendations for future research in mental health. *Gen. Hosp.* 35, 4 (2013), 332-338.
- [12] Sommerville, I. 2007. *Software Engineering* (9th ed.). Addison-Wesley Publishing Company, Boston, MA.
- [13] Marr, D. 2010. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. The MIT Press, Cambridge, MA.
- [14] Fitzpatrick, K. K., Darcy, A., and Vierhile, M. 2017. Delivering Cognitive Behavior Therapy to Young Adults With Symptoms of Depression and Anxiety Using a Fully Automated Conversational Agent (Woebot): A Randomized Controlled Trial. *JMIR Ment. Health.* 4, 2 (2017), e19.
- [15] Provoost, S., Lau, H. M., Ruwaard, J., and Riper, H. 2017. Embodied Conversational Agents in Clinical Psychology: A Scoping Review. *J. Med. Internet. Res.* 19, 5 (2017), e151.
- [16] Yorita, A., Egerton, S., Oakman, J., Chan, C., and Kubota, N. 2018. A Robot Assisted Stress Management Framework. *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Miyazaki, Japan, 2018*.
- [17] Fulmer, R., Joerin, A., Gentile, B., Lakerink, L., and Rauws, M. 2018. Using Psychological Artificial Intelligence (Tess) to Relieve Symptoms of Depression and Anxiety. *JMIR Ment. Health.* 5, 4 (2018), e64.
- [18] Stodden, V. 2011. Trust your science? *Amstat News*.
- [19] Vinyals, O. and Le, Q. 2015. A neural conversational model. In *Proceedings of the ICML Deep Learning Workshop 2015* (Lille, France, July 10-11, 2015). IMLS, Lille, France.
- [20] Sosnovsky, S. and Dicheva, D. 2010. Ontological technologies for user modeling. *Int. J. Metadata Semant. Ontol.* 5, 2 (2010), 32-71.
- [21] Sunstein, C. and Thaler, R. 2008. *Nudge: Improving Decisions about Health, Wealth, and Happiness*. Yale University Press, New Haven, Connecticut.
- [22] Rammstedt, B. and John, O. P. 2007. Measuring personality in one minute or less. *J. Res. Pers.* 41, 1 (2008), 203-212.
- [23] IBM Personality models. 2019. *IBM Cloud*.
- [24] Soldz, S. and Vaillant, G. E. 1999. The Big Five personality traits and the life course: A 45-year longitudinal study. *J. Res. Pers.* 33 (1999), 208-232.
- [25] Lovibond, S. H. and Lovibond, P. F. 1995. *Manual for the Depression Anxiety Stress Scales (DASS)*. Psychology Foundation, Sydney.
- [26] Fenza, G., Orciuoli, F., and Sampson, D. G. 2017. Building adaptive tutoring model using artificial neural networks and reinforcement learning. In *17th International Conference on Advanced Learning Technologies* (Timisoara, Romania, July 3-7, 2017). IEEE, 460-462.
- [27] Bučar, J. 2017. *Slovene sentiment lexicon JOB 1.0, Slovenian language resource repository CLARIN.SI*. <http://hdl.handle.net/11356/1112>.
- [28] Coppersmith, G., Dredze, M., and Harman, C. 2014. Quantifying mental health signals in Twitter. In *Proceedings of the Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality* (Maryland, USA, June, 2014). ACL, 51-60.
- [29] Gjoreski, M., Luštrek, M., Gams, M., and Gjoreski, H. 2017. Monitoring stress with a wrist device using context. *J. Biomed. Inform.* 73 (2017), 159-170.
- [30] Wen, W. H. et al. 2018. Toward constructing a real-time social anxiety evaluation system: Exploring effective heart rate features. *IEEE T. Affect. Comput.* Early access (2018).
- [31] Fedor, S., Ghandeharioun, A., Picard, R., and Ionescu, D. 2019. *U.S. Patent Application No. 16/168, 378*.
- [32] Bolliger, L., Lukan, J., Luštrek, M., Bacquer, D. D. and Clays, E. 2019. Disentangling the Sources and Context of Daily Work Stress: Study Protocol of a Comprehensive Real-Time Modelling Study Using Portable Devices. *World Acad. Sci. Eng. Technol.* 13, 3 (2019), 3755.
- [33] Gams, M. 2001. *Weak intelligence: through the principle and paradox of multiple knowledge*. Nova Science, Hauppauge, NY.
- [34] Clark, A. 2013. Whatever next? Predictive brains, situated agents, and the future of cognitive science. *Behav. Brain Sci.* 36 (2013), 181-204.
- [35] Mlakar, M., Tavčar, A., Grasselli, G., and Gams, M. *Asistent za stres*. <http://poluks.ijs.si:12345/>.

Classifying Power Quality Disturbances in Noisy Conditions using Machine Learning

Bojana Velichkovska

Faculty of Electrical Engineering
and Information Technologies
Ss. Cyril and Methodius
University
1000 Skopje, N. Macedonia
bojanav@feit.ukim.edu.mk

Marija Markovska

Faculty of Electrical Engineering
and Information Technologies
Ss. Cyril and Methodius
University
1000 Skopje, N. Macedonia
marijam@feit.ukim.edu.mk

Hristijan Gjoreski

Faculty of Electrical Engineering
and Information Technologies
Ss. Cyril and Methodius
University
1000 Skopje, N. Macedonia
hristijang@feit.ukim.edu.mk

Dimitar Tashkovski

Faculty of Electrical Engineering and Information Technologies
Ss. Cyril and Methodius University
1000 Skopje, N. Macedonia
dtaskov@feit.ukim.edu.mk

ABSTRACT

When ensuring high-quality power supply of the power grid it is of the utmost importance to correctly detect and classify any power quality (PQ) disturbance. Selecting the most relevant features is very important in the process of training a general machine learning model. Therefore, we analyze the power signals and extract information from them, and then select the most significant features. Additionally, an effective classification model is required. In this study we apply grid search throughout the features sets on one side, and the classification algorithms on the side. This way, we determine the most effective combination of an algorithm and feature set for classification of power quality disturbances.

Keywords

Power quality, feature extraction, classification, machine learning.

1. INTRODUCTION

As nowadays the power grid undergoes many disturbances, it has become more of a challenge to not only detect these disturbances, but also to correctly identify them (as harmonics, transients, voltage dips, etc.). Awareness regarding a disruption within a signal could be used to minimize the undesired effects which poor power quality could inflict. The information gathered during detection and classification of power grid malfunctions could be utilized as insight in solving them. Additionally, the reliability of a power distribution network is key in preventing complaints by customers, which is why there is an increased focus on detection of disturbances in the grid.

In the recent years, with the development of the novel machine learning and feature extraction techniques, the power quality disturbances domain have also benefitted by introducing various approaches based on machine learning.

With regard to the feature extraction methods used to formulate the data frame before applying an algorithm, there are several digital signal processing techniques. S-transform is one of the most commonly used feature extraction approaches [1]. Other popular feature extraction tools are short-time Fourier transform (STFT), fast working Fourier transform (FT), Neural Networks, Wavelet Transform (WT) and Discrete Wavelet Transform [2][3][4][5]. WT analyses give frequency and time information accurately by convolving the dilated and translated wavelet with the input signal. This property makes the WT approach suitable for detecting various deviations in voltage and current waveform, caused by different PQ disturbances.

Proposed methods for classifications of PQ disturbances include rule-based approaches, as the one presented in [1], time series analysis [5], artificial neural networks (ANNs) [6] and machine learning techniques as Support Vector Machines, Decision Trees and Random Forest [7][8].

A similar study to ours, explained in [9], shows results upon applying machine learning algorithms to PQ signals, where no phase shifts occur. Additionally, only 11 classes are being processed. The trained models achieve at the highest an accuracy of 98.38% in pure signals, by using combinations of features based on discrete wavelet transform. Our data has the included difficulty of phase shifts in 21 different class and a subset of data where the presence of noise is not labeled by exact value. Another approach is shown in [4], where neural network perform the classification of signals based on features extracted with discrete wavelet transform. The results obtained in this paper give accuracies displayed by class, which range from 76% to 100%. Overall, the approach can perform with equal quality as the energy difference patterns remain same in noisy environments, but the obtained model from the research can detect a total of 11 different types of disturbances, whereas we have 21 classes investigated.

Upon combining different feature extraction methods with machine learning approaches unique investigations of the problem are

provided. We utilize the results of feature extraction and feature selection design to create descriptive feature groups and to use the obtained groups for training various models. The feature extraction process carries significant weight in addressing the type of disturbance inhibiting the power signal. A correctly determined set of features can be crucial for the accuracy of a model, as on a weak set of features even the best approach would provide extinguishable results. Moreover, a compressed feature set can influence the required processing time.

The processing of the signals and the feature extraction and selection method are addressed in section II, the approach methods are discussed in section III, the results are presented in section IV and we conclude this research in section V.

2. DATA AND FEATURE EXTRACTION

Our dataset is comprised of samples from 21 different type of PQ disturbances, as given in Table I, generated in accordance with the mathematical definitions given in [9]. We used those samples as PQ signals accompanied with 20 dB, 30 dB, 40 dB and 50 dB white Gaussian noise. Every signal we examined contained 10 cycles, for fundamental frequency of 50 Hz and sampling frequency of 3.2 kHz. Accordingly, a signal instance in our dataset contained 640 data samples linked in a time series. Note that there were 21000 training and 21000 test signal instances.

Table 1: PQ Classes

PQ Disturbance	Classes
Pure	C1
Sag	C2
Swell	C3
Interruption	C4
Transient/Impulsive/Spike	C5
Oscillatory transient	C6
Harmonics	C7
Harmonics + Sag	C8
Harmonics + Swell	C9
Flicker	C10
Flicker + Sag	C11
Flicker + Swell	C12
Sag + Oscillatory transient	C13
Swell + Oscillatory transient	C14
Notch	C15
Harmonics + Sag + Flicker	C16
Harmonics + Swell + Flicker	C17
Harmonics + Sag + Oscillatory transient	C18
Harmonics + Swell + Oscillatory transient	C19
Harmonics + Sag + Flicker + Oscillatory transient	C20
Harmonics + Swell + Flicker + Oscillatory transient	C21

Upon these signal instances we preformed automatic extraction of relevant features with the help of tsfresh (Time Series FeatuRe Extraction on basis of Scalable Hypothesis tests) [10]. The tsfresh

library is used to accelerate the process of extracting features by combining 63 time series characterization methods, which resulted in 794 time series features. Related work that use tsfresh for feature extraction is proposed in [12].

The obtained features can be applied for classification. We used the complete set of extracted features that tsfresh provides, to train our models. However, as the number of obtained features was significant, it was necessary to influence the computational efficiency of the models. To do so, we used a feature selection module designed to reduce the number of features by selecting the most relevant descriptors of the problem.

Firstly, we determined the mutual information between each extracted feature and the corresponding class. Considering the fact that the higher the value of the mutual information the more relevant the feature is to the class, we sorted the features in descending order based on this value. We selected the upmost 300 features. Next, we divided the features in groups of 50. For the first group, we calculated the Pearson correlation coefficient [11] for every pair of features. Upon encountering a pair with a correlation higher than 0.8, the feature with the lower mutual information to the class was removed. Once the whole group was iterated, the next group was appended to the remainder of the features and the process was repeated for all groups, until a final set of features was obtained.

These features were a mix of both time domain features and frequency domain features. We inputted the time domain and frequency domain features to our models separately, as well as a union between the sets upon training our models.

3. MACHINE LEARNING APPROACH

With the four feature-wise possible data frames (all initial features, remainder of features after the feature selection process only containing time domain features, only containing frequency domain features and a combination of the two) we trained our models with 1000 signals per class. The signals used for training and testing within one class had different phase shifts. Our test set also contained 1000 signals per class.

We trained our models with different signal sets, where each set was synthesized with 20 dB, 30 dB, 40 dB or 50 dB of white Gaussian noise. Also, a set of signals synthesized with 20-50 dB of noise was used. The latter is due to the noise variations occurring within a real signal. The following five algorithms were used for the experimental analysis.

- *K Nearest Neighbors (KNN)* is algorithm that analyzes a test sample in comparison to the whole of the train data. Through comparing the sample with the whole of the data frame, the closest neighbors which have the highest similarity with the sample are found. They determine the class in which the sample will be placed.
- *Decision Tree* is an algorithm which preforms data division by splitting the frame into several branches recursively, in a way that each split is determined by a value of one feature from the data frame. The branching ends when a class for the analyzed sample is reached.
- *Random Forest* is an ensemble algorithm of decision trees. Each tree performs on the values of a random feature vector sampled

independently and with the same distribution for all trees in the forest.

- *Gradient Boosting* produces a prediction model in a stage-wise fashion as an ensemble of weak prediction models, typically decision trees.
- *XGBoost* is an implementation of gradient boosted decision trees which are designed for speed and performance.

4. RESULTS

Through the results given in Tables 2-6, we can observe the behavior of our algorithms depending on the noise and the feature set selected. We concluded that using all the features tsfresh extracts to train our models provided the highest accuracy. After we extracted the features with the method described in Section 2 and trained our models with the extracted features, a significant decrease in the accuracy of the models occurs. Those remaining features we divided into two categories, time domain and frequency domain features, before we retrained our algorithms. The time domain features in the worst-case scenario provided insignificant decrease in the accuracy of the classifiers, whereas the accuracy obtained through the frequency domain features caused a severe drop in accuracy.

When analyzing the algorithms themselves it shows that on overall the XGBoost algorithm performed best.

Table 2: Comparison of algorithms accuracy with 20 dB noise.

20 dB white Gaussian noise				
Classifier	Extracted Features	Selected Features	Time Domain Features	Frequency Domain Features
Nearest Neighbors	0.99	0.43	0.46	0.31
Decision Tree	0.29	0.6	0.54	0.37
Random Forrest	0.43	0.42	0.53	0.26
Gradient Boosting	0.99	0.75	0.75	0.53
XGBoost	0.98	0.78	0.77	0.54

Table 3: Comparison of algorithms accuracy with 30 dB noise.

30 dB white Gaussian noise				
Classifier	Extracted Features	Selected Features	Time Domain Features	Frequency Domain Features
Nearest Neighbors	0.99	0.46	0.52	0.34
Decision Tree	0.17	0.68	0.63	0.43
Random Forrest	0.48	0.52	0.6	0.25
Gradient Boosting	0.98	0.85	0.85	0.55
XGBoost	0.98	0.88	0.87	0.57

Table 4: Comparison of algorithms accuracy with 40 dB noise.

40 dB white Gaussian noise				
Classifier	Extracted Features	Selected Features	Time Domain Features	Frequency Domain Features
Nearest Neighbors	0.99	0.54	0.59	0.43
Decision Tree	0.29	0.68	0.65	0.5
Random Forrest	0.43	0.57	0.6	0.39
Gradient Boosting	0.99	0.87	0.87	0.63
XGBoost	0.99	0.9	0.89	0.65

Table 5: Comparison of algorithms accuracy with 50 dB noise.

50 dB white Gaussian noise				
Classifier	Extracted Features	Selected Features	Time Domain Features	Frequency Domain Features
Nearest Neighbors	0.99	0.57	0.63	0.45
Decision Tree	0.29	0.63	0.65	0.52
Random Forrest	0.51	0.58	0.69	0.44
Gradient Boosting	0.98	0.87	0.87	0.64
XGBoost	0.99	0.9	0.89	0.7

Table 6: Comparison of algorithms accuracy with 20-50 dB noise.

20-50 dB white Gaussian noise				
Classifier	Extracted Features	Selected Features	Time Domain Features	Frequency Domain Features
Nearest Neighbors	0.98	0.51	0.57	0.4
Decision Tree	0.42	0.64	0.6	0.43
Random Forrest	0.45	0.54	0.59	0.3
Gradient Boosting	0.99	0.85	0.84	0.6
XGBoost	0.99	0.88	0.87	0.6

5. CONCLUSION

With this paper we addressed the effectiveness of observing the PQ signals as time series and using features obtained with tsfresh in classification of PQ disturbances. The testing was conducted with numerous classification models, each time on 21 classes and accompanied with different noise levels.

The results show that features extracted with tsfresh can be used for correctly classifying PQ disturbances. However, a deeper understanding in choosing subsections of those features is needed, and also additional testing.

Our future endeavors will be creating a more optimal feature set, which might consist of a combination of our current features and features obtained from Wavelet Transform. Additionally, we will investigate the possibility of improvements using deep learning methods.

6. ACKNOWLEDGMENTS

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

7. REFERENCES

- [1] Rodríguez, A & Aguado, Jose & Martín, F & Lopez, J.J. & Muñoz, F & Ruiz, J.E.. (2012). Rule-based classification of power quality disturbances using S-transform. *Electric Power Systems Research*. 86. 113–121. 10.1016/j.epsr.2011.12.009.
- [2] I. Daubechies, *Ten lectures on wavelets*, Philadelphia, PA: Society for Industrial and Applied Mathematics, (1992).
- [3] C.S. Burrus, R.A. gopinath, and H.Guo, *Introduction to Wavelets and Wavelet Transforms: A Primer* (Englewood Cliffs, NJ: Prentice-Hall, 1998).
- [4] Subhamita Roy, and Sudipta Nath, Classification of power quality disturbances using features of signals, *International Journal of Scientific and Research Publications*, Vol. 2, No. 11, November 2012,01-09.
- [5] Lalit Kumar Behera, Maya Nayak, and Sareeta Mohanty, Discrete wavelet transform and S-transform based time series data mining using multilayer perception neural network, *International Journal of Engineering Science and Technology*, Vol. 3 No. 11, November 2011, 8039-8046.
- [6] Monedero, Iigo & León, Carlos & Roperro, Jorge & García, Antonio & Manuel Elena, Jos & Montaña, Juan-Carlos. (2007). Classification of Electrical Disturbances in Real Time Using Neural Networks. *Power Delivery, IEEE Transactions on*. 22. 1288 - 1296. 10.1109/TPWRD.2007.899522.
- [7] Milchevski, A.; Taskovski, D. Classification of power quality disturbances using wavelet transform and SVM decision tree. In *proceedings of the 11th IEEE Electrical Power Quality and Utilization Conference, Lisbon, Portugal, 17–19 October 2011*.
- [8] Markovska, Marija & Taskovski, Dimitar. (2018). The effectiveness of wavelet based features on power quality disturbances classification in noisy environment. 10.1109/ICHQP.2018.8378873.
- [9] Markovska, M.; Taskovski, D. On the choice of wavelet based features in power quality disturbances classification. In *Proceedings of the 2017 IEEE 17th International Conference on Environment and Electrical Engineering (EEEIC), Milan, Italy, 6–9 June 2017*.
- [10] M. Christ, A.W. Kempa-Liehr, M. Feindt, Distributed and parallel time series feature extraction for industrial big data applications. *Asian Machine Learning Conference (ACML)* 2016, Workshop on Learning on Big Data (WLBD), Hamilton (New Zealand), ArXiv preprint arXiv: 1610.07717v1.
- [11] F. Galton, Co-relations and their measurement, chiefly from anthropometric data. *Proceedings of the Royal Society of London* 45 (1888):135–145. ACM Woodstock conference.
- [12] Janko, Vito & Lustrek, Mitja & Reščič, Nina & Mlakar, Miha & Drobnič, Vid & Gams, Matjaz & Slapničar, Gašper & Gjoreski, Martin & Bizjak, Jani & Marinko, Matej. (2018). A New Frontier for Activity Recognition: The Sussex-Huawei Locomotion Challenge. 1511-1520.

EMZ – Elektronsko in mobilno zdravstvo

Aleš Tavčar
Institut "Jozef Stefan"
Jamova cesta 39,
1000 Ljubljana
ales.tavcar@ijs.si

Jani Bizjak
Institut "Jozef Stefan"
Jamova cesta 39,
1000 Ljubljana
Jani.bizjak@ijs.si

Matjaž Gams
Institut "Jozef Stefan"
Jamova cesta 39,
1000 Ljubljana
matjaz.gams@ijs.si

POVZETEK

Področje elektronskega in mobilnega zdravja ali e-zdravje postaja v zadnjih letih ključno za spopadanje z izzivi v zdravstvu, staranjem prebivalstva in zagotavljanje ustrezne ter celovite zdravstvene oskrbe. V okviru programa Pametne specializacije S4 EkoSMART in projekta EMZ smo želeli vzpostaviti okolje IKT, ki bi služilo informatizaciji področja zdravja in državljanom ponudilo nove platforme, storitve ter produkte, kar bi jim omogočalo celovitejšo oskrbo in hitrejši dostop do zdravstvenih storitev. S tem pa bi pripomogli k izboljšanju zdravstvene oskrbe in hitrejši obravnavi bolnikov v Sloveniji. V tem prispevku opravimo pregled izvedenih razvojnih aktivnosti in razvitih prototipov v okviru projekta EMZ.

ABSTRACT

The field of electronic and mobile health or eHealth has gained in importance in recent years to meet the challenges of healthcare, the aging of the population, and the provision of adequate and comprehensive health care. As part of the Smart Specialization S4 EkoSMART program and the EMZ project, we wanted to create an ICT ecosystem that would improve the health field and offer citizens new platforms, services, and products that would enable them to provide more comprehensive care and faster access to health services. In this paper, we review the development activities performed, and the prototypes developed within the EMZ project.

Ključne besede

EMZ, e-zdravje, jezikovne tehnologije

1. UVOD

V pričujočem preglednem prispevku podamo pregled opravljenih razvojnih in raziskovalnih aktivnosti na projektu EMZ – Elektronsko in mobilno zdravstvo, ki je eden izmed petih projektov krovnega programa EkoSMART [1] (2016-2019). To je program Slovenske strategije pametne specializacije (S4) za področje Pametna mesta in skupnosti. V projektu EMZ je sodelovalo 15 partnerjev. Glavni namen projekta je bil vzpostavitev okolja IKT za podporo področju zdravja in znotraj tega razvoj prototipov naprednih rešitev za klinično in domačo zdravstveno oskrbo.

Aktivnosti znotraj projekta EMZ lahko v grobem razdelimo na štiri podpodročja:

- (i) definiranje Nacionalne strategije EMZ;
- (ii) razvoj komunikacijske EMZ platforme;
- (iii) razvoj IKT prototipov storitev ali produktov za področje zdravja;
- (iv) verige vrednosti.

Definiranje Nacionalne strategije EMZ ni mišljeno kot izdelava programskega sistema, ampak je namenjeno odločevalcem (državnemu zboru, ministrstvu) kot priročnik za digitalizacijo in informatizacijo zdravstva. Komunikacijska EMZ platforma je namenjena splošni populaciji in njena glavna značilnost je

enostavnost uporabe, saj omogoča komunikacijo v naravnem jeziku in ponuja intuitiven uporabniški vmesnik. Poleg tega integrira različne storitve, ki so bile razvite v okviru projekta EMZ drugih projektov in omogoča posredovanje informacij o razvitih produktih ter prototipih. Glavni del aktivnosti znotraj projekta EMZ pa je bil posvečen razvoju IKT prototipov za področje zdravja. Spričo velikosti ter večanja populacije starejših in kroničnih bolnikov in njihovih specifičnosti potreb je bil namen posameznih partnerjev izdelati sisteme, ki izboljšujejo varstvo in varnost široke skupine ljudi: starejših, kroničnih bolnikov, ki občasno ali stalno potrebujejo zdravniško oskrbo, ljudi s posebnimi potrebami in tudi zdravih oseb z namenom preventive. Verige vrednosti pa predvidevajo povezovanje različnih deležnikov in sodelovanje pri razvoju prototipov za različna področja, ki zahtevajo natančno specificirano obravnavo.

Doseganje teh ciljev (podpodročij) smo želeli v okviru EMZ doseči z uporabo modernih IKT tehnologij in pristopov. Razviti sistemi bodo stremeli k nadgradnji kakovosti današnje obravnave in oskrbe omenjenih skupin s pomočjo elektronskih in mobilnih naprav in pripadajočih programskih rešitev, zlasti umetne inteligence v oblaku ali lokalno, npr. na mobilni napravi ali s prilagojenimi senzorji in nosljivimi pripomočki. Namen EMZ je tudi, da ob tem nastajajo celovite verige vrednosti, ki bodo uspešno vstopile v nišne prostore na mednarodnih trgih.

V tem preglednem prispevku predstavimo opravljene razvojno-raziskovalne aktivnosti na različnih področjih, ki jih pokriva projekt ter navedemo nekaj prototipov rešitev, ki so jih partnerji razvili. Razviti prototipi spadajo v eno od skupin storitev/produktov/izdelkov, ki so vsebinsko razdeljene v različna področja

2. SPLOŠNO O EMZ

Projekt EMZ se je izvajal v okviru programa EkoSMART (S4 – Pametna mesta in skupnosti) in ga je koordiniral Odsek za inteligentne sisteme, Institut »Jozef Stefan«. EMZ je pretežno razvojno-raziskovalne narave in glavni rezultati aktivnosti na projektu so namenjeni razvoju prototipov novih produktov, aplikacij in storitev, ki bi lahko bili po preverbi ustreznosti prenešeni na trg preko zainteresiranih partnerjev, na pa tudi neposredno tržnim sistemom. EMZ je razdeljen na več delovnih paketov, kjer so aktivnosti v prvih štirih zasnovane tako, da pokrivajo ločena področja EMZ in so relativno neodvisna ena od druge. Del pa je zasnovan povezovalno, saj je usmerjen k razvoju povezovalne platforme. Pregled delovnih paketov je podan spodaj:

- **DP1: Informacijske tehnologije za podporo celostni oskrbi / bolnice.** Delovni paket se osredotoča na informacijske tehnologije, ki so namenjene celostni oskrbi.
- **DP2: Podpora na domu za zdrave, starejše in za kronične bolnike / doma.** Fokus aktivnosti je na izboljšanju obstoječih in razvoju novih prototipov, ki so

namenjeni podpori na domu za starejše, kronične bolnike in zdrave.

- **DP3: Mobilno spremljanje vitalnih in okoljskih podatkov / mobilno.** Delovni paket predvideva razvoj rešitev, ki so povezane z mobilnim spremljanjem vitalnih in okoljskih parametrov.
- **DP4: Računalniška podpora, podatki, kreiranje novih znanj / algoritmi.** Aktivnosti se osredotočajo na uporabo naprednih algoritmov za analizo podatkov, procesiranje podatkov iz naprav, podpori storitev za uporabnike, analizi slik in kreiranju novih znanj.
- **DP5: IKT platforma.** Obsega razvoj povezovalne platforme, ki omogoča povezljivost naprav ter senzorje, skupni zajem ter hranjenje meritev.

Partnerji po delovnih paketih stremimo k razvoju lastnih prototipov, ki bi lahko bili tržno zanimivi. Pri tem pa se osredotočamo na povezljivih sistemov, kar bi lahko po zaključku projekta omogočalo integracijo v skupno integracijsko platformo pametnega mesta, ki se razvija v okviru programa EkoSMART.

3. RAZVOJNE AKTIVNOSTI

Ta razdelek podrobneje prestavi opravljene aktivnosti na štirih glavnih podpodročjih projekta EMZ (glej Uvod). Najprej predstavimo predvideno vsebino nacionalne strategije EMZ, sledi opis komunikacijske platforme EMZ, nato predstavimo glavne prototipe storitev/produktov/izdelkov, ki so vsebinsko razdeljene v različna področja: rehabilitacijska orodja, pametne naprave za spremljanje oseb, naprave za spremljanje življenjskih funkcij, pametne naprave ter storitve EMZ in platforme ter spletne aplikacije. V zaključku razdelka pa navedemo opravljene aktivnosti povezane z vzpostavitvijo verig vrednosti.

3.1 Nacionalna strategija EMZ

Osnovna teza vizije EMZ v Sloveniji je, da je digitalizacija zdravstva še v začetnih fazah. Sistematična tranzicija na informacijske tehnologije bi lahko pohitрила obravnavo bolnikov, izboljšala kvaliteto storitev in zmanjšala stroške. Ker pa je bila sedanja uporaba tehnologij IKT v Sloveniji problematična, saj je pogosto povzročala dodatne zamude in birokratizacijo, je potrebno presekati s takim neproduktivnim pristopom. Hkrati je v Sloveniji veliko nekompatibilnih sistemov, v UKC kar 7. Prav tako še vedno ni vzpostavljene infrastrukture za učinkovito vpeljavo telemedicine na domu. S tem bi lahko zmanjšali čakalne dobe, povečali zmožnost obravnave bolnikov in obenem zmanjšali pritisk na zdravstveno osebje in celoten zdravstveni sistem. Napovedi strokovnjakov so (v svetovnem okviru), da se bo premik dogodil v roku desetletja. V okviru definiranja nacionalne strategije EMZ bomo pripravili Belo knjigo EMZ. Predlog bo predstavljen tako Ministrstvu za zdravje kot na posvetu v Državnem svetu v roku nekaj mesecev.

3.2 Komunikacijska platforma EMZ

Komunikacijska EMZ platforma je namenjena splošni populaciji. Njena glavna značilnost je enostavnost uporabe, saj omogoča komunikacijo v naravnem jeziku in ponuja intuitiven uporabniški vmesnik. Platforma je bila razvita za namenom povezovanja in integracije različnih produktov in storitev, ki so bili razviti v okviru projekta EMZ in širše. Platforma temelji na odprtokodni rešitvi Rocket.Chat [8] in omogoča uporabnikom komunikacijo v naravnem jeziku in s tem posredovanje informacij iz različnih domen.

Platforma ima tri glavne funkcionalnosti: (i) posredovanje odgovorov na vprašanja uporabnikov, ali pa posredovanje povezav

do informacij, ki jih uporabnik potrebuje; (ii) pridobivanje informacij ali podatkov iz zunanjih spletnih servisov; (iii) iskanje po zbirkah podatkov vključenih institucij, tako da za iskanje v bazi podatkov najprej opravi stavčno analizo uporabniškega zahtevka in nad določene termine aplicira lematizacijo, pridobljene koncepte nato uporabi za identifikacijo ustreznih storitev ali terminov. Platforma pridobiva odgovore iz množice pogovornih svetovalcev, ki obravnavajo različne domene in so integrirani v platformo. V nadaljevanju na kratko predstavimo potek pridobivanja odgovorov in način delovanja svetovalnih agentov.

Uporabnik postavi vprašanje, ki ga sistem procesira. Na začetku se opravi enostavno zaznavanje namena. Če se tematika vprašanja nanaša na področja, ki jih obravnavajo svetovalni agenti in ne na bazo prototipov in domen (množica podatkovnih zbirk), glavni agent platforme posreduje celoten niz besedila svetovalcem, ki so integrirani v platformo. V trenutni različici so to Izbirčič, asistent za čakalne vrste, EkoSMART asistent in občinski asistenti. Slednji nato izvedejo naprednejše procesiranje teksta: stavčno analizo, lematizacijo, zaznavanje namena in luščenje ključnih podatkov. Nato se v bazi znanja poišče najboljši odgovor, ki se posreduje nazaj platformi. V primeru, da vprašanje zadeva tematiko prototipov in domen, agent odgovor razbere neposredno iz baze podatkov. Pred tem glavni agent platforme izvede luščenje iskalnih konceptov in z njimi opravi iskanje v podatkovni bazi. Podatkovna zbirka domen in prototipov se redno posodablja. V nadaljevanju zberemo vse odgovore v tabelarično podatkovno strukturo. Odgovori se filtrirajo in izberejo samo tisti, za katere glavni agent oceni, da so smiselni. Na koncu se jih prikaže v pogovornem oknu platforme. Uporabnik lahko v oknu izbere najustreznejši odgovor. S klikom na gumb lahko nadaljuje pogovor s svetovalcem, ki je posredoval izbrani odgovor. Ko uporabnik to stori, drugi svetovalci ne odgovarjajo.

3.3 Prototipi EMZ

V nadaljevanju predstavimo nekaj izbranih prototipov izmed več kot 21, ki so bili razviti v okviru projekta EMZ. Najprej opravimo pregled prototipov, ki so bili razviti s strani partnerjev in nato še izpostavimo tri prototipe, ki so bili razviti na IJS.

3.3.1 Partnerji

V okviru projekta EMZ je bilo s strani partnerjev razvitih več prototipov, ki so v višjih fazah razvoja in so tržno zanimivi. V nadaljevanju podamo 6 primerov takšnih prototipov in jih na kratko opišemo.

Izvajal se je razvoj nove zapestnice za detekcijo padcev in beleženje aktivnosti starostnikov. Aktivnosti so vključevale razvoj strojne opreme za prototipno napravo, ki omogoča testno zbiranje podatkov, ki bodo uporabljeni za naprednejše algoritme. Izdelana je bila priključna shema ter načrt za izvedbo tiskanega vezja ter za polaganje elementov. Izdelano je bilo prototipno vezje ter programska oprema, ki omogoča testiranje izbranih komponent. Glavne funkcionalnosti naprave so spremljanje aktivnosti uporabnikov preko pospeškometra, zaznavanje lokacije uporabnika, gumb za klic v sili in glasovni klic ob izrednih dogodkih (padeč).

Za področje rehabilitacije se je razvil prototip 10Cubes namenjen terapiji bolnikov (Parkinsonova bolezen, možganska kap ipd.). Gre za programsko opremo, kjer je 10 raznobarnih kock naključno razporejenih po navideznem travnatem prostoru, uporabnik pa jih mora ena po ena zložiti v odprto skrinjo. Celotno okolje je bilo razvito v okolju Unity3D, informacijo o gibanju roke in prstov (kinematiko) pa se zajema s komercialno 3D kamero LeapMotion. Razviti so bili napredni moduli za analizo gibanja roke in prstov.

Vsak članek, dlan in zapestje imajo vpet koordinatni sistem, le-tega pa se poveže v kinematično verigo. Z matičnim računom se tako izračuna položaj vsakega segmenta v 3D prostoru.

Naslednji prototip je bluetooth naprava (značka, zapestnica) za lokalizacijo bolnikov ali drugih oseb v prostoru. Sistem je izveden s fiksnim senzorjem na ploščici Raspberry PI in napravami, ki oddajajo signal. Senzor zazna napravo v prostoru, jo identificira, analizira signal in ga shrani v bazo podatkov.

V okviru projekta je potekal nadaljnji razvoj mobilnega EKG z visoko ločljivostjo zapisa. Nadgradnje so potekale v smeri izboljšanja zaznavanja in beleženja ekstremno majhnih biopotencialov. V ta namen so se modificirale vhodne naprave in tako uspešno neinvazivno in pasivno beležilo električno aktivnost ploda v maternici. Aktivnosti so se fokusirale tudi v odpravljanje motenj, ki jih povzročata »elektrosmog« in s tem se izboljša samo ločljivost naprave. Izvedene so bile integracije z različnimi platformami in sistemi partnerjev: vmesniki do Think!EHR [4] (zdravstvena platforma podjetja Marand, ki temelji na EHR zapisih), komunikacijski protokol za prenos podatkov v platformo, prikaz EKG poročil in meritev v zdravstvenem informacijskem sistemu (BIRPIS32 [7]).

Potekal je razvoj prototipa pametne ščetke. Osnovna ideja je opremiti zobno ščetko s senzorji, ki naj omogočajo detekcijo gibanja (pospeškomer) in merjenje pritiska ščetke na zobe. Zbrane podatke se prenese v oblak, kjer se podatki analizirajo in pridobijo ustrezne informacije. Razviti ter povezani so bili ustrezni mehanski deli, pospeškometer in pretvornik signalov iz merilnih lističev za merjenje pritiska. Senzorske signale ob uporabi ščetke (pospeški in sila) sprejema Arduino in jih preko protokola BLE (Bluetooth low energy) prenaša na mobilne naprave. Implementiran je bil tudi prototip resne igre za vzdrževanje zobne higiene, ki je povezan s prototipom ščetke in izvedene gibe ščetke preslika v akcije v igri.

Razvil se je laboratorijski prototip lahke IoT platforme, ki temelji na odprtih M2M in IoT standardih in omogoča komunikacijo z napravami in senzorji prek vseh ključnih komunikacijskih protokolov, vključno z BT, WiFi, CoAP, MQTT, 6LowPAN, ZigBee, ZWawe. Poleg celotnega sklada protokolov platforma omogoča tudi zajem in procesiranje podatkov za potrebe detekcije kompleksnih dogodkov in realizacije lokalne inteligence (npr. na nivoju stavbe ali oskrbovanega stanovanja).

3.3.2 IJS

V tej sekciji predstavimo tri prototipe storitev in izdelkov, ki so bile razvite na Odseku za Inteligentne sisteme.

Nadaljevanje razvoja in nadgradnje prototipa programske opreme za androidne pametne ure, ki samostojno zaznava padce in ima 15 funkcij. Aplikacija skrbi za varnost uporabnika s stalnim nadzorom ter procesiranjem signalov iz senzorjev in obveščanjem v primeru izrednih dogodkov kot so padci. Razvoj je vključeval nadgradnjo aplikacije za pametno uro, vključitev algoritmov strojnega učenja za učinkovito zaznavanje padcev, optimizacija delovanja z namenom varčevanja z baterijo. Vključeno je bilo posredovanje meritev in dogodkov v spletni servis, ki podatke analizira in omogoča pregleden prikaz nadzornikom sistema. Dodana je možnost geo-lociranja in nastavljanja preko spleta ali sms sporočil. Glavni del sistema je aplikacija, ki spremlja aktivnosti uporabnika. V primeru padca se sproži avtomatski klic na pomoč. Če se na klicani strani nihče ne odzove, ura ponovno kliče vse številke, ki so vnesene v seznam skrbnikov.

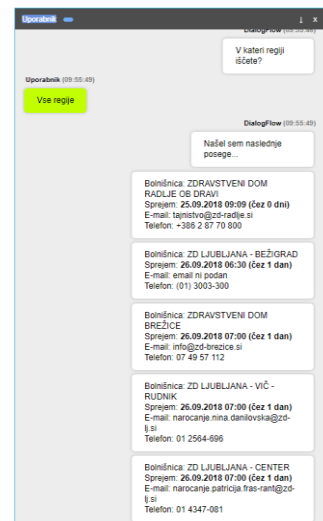
V okviru EMZ smo razvili več aplikacij in storitev, ki temeljijo na jezikovnih tehnologijah in procesiranju naravnega jezika z

namenom komunikacije z uporabniki in posredovanja informacij. V nadaljevanju predstavimo dve. Prva tovrstna storitev je aplikacija za posredovanje informacij o čakalnih dobah za posamezne zdravstvene storitve.

Splošna arhitektura predlaganega sistema je sestavljena iz štirih glavnih komponent:

- spletnega vmesnika za zajem uporabniških poizvedb,
- pogovornega agenta znotraj sistema DialogFlow [2] za procesiranje jezika,
- zalednega mehanizma, ki procesira zahtevke iz DialogFlow agenta,
- univerzalnega vmesnika za dostop do podatkovnih zbirk.

Spletni vmesnik je implementiran v obliki pogovornega okna (angl. chat), kjer se vprašanja in odgovori izmenično prikazujejo (glej sliko 1). Odgovori sistema se prikazujejo na desni strani in so osenčeni s sivo barvo. Vnosi uporabnika pa se prikazujejo na levi strani in so osenčeni s svetlo zeleno barvo. Vmesniku so dodani tudi izbirni gumbi, kar olajša uporabniku interakcijo, saj mu ni potrebno pisati in lahko izbere zeleno opcijo.



Slika 1: Prikaz rezultatov poizvedovanja za zobozdravstveni pregled.

Procesiranje tekstovnih zahtev uporabnika poteka preko storitve DialogFlow za prepoznavanja namena (angl. intent recognition), ki klasificira zahtevke v različne kategorije. Na primer, uporabnik lahko povpraša po določeni zdravstveni storitvi ali pa poda nujnost napotnice. Svetovalec najprej poskuša razbrati iz vnosov uporabnika, za kateri zdravstveni poseg se zanima. To je realizirano s kombiniranjem algoritma za indeksiranje izrazov in klasifikacijo ključnih konceptov. V DialogFlow agentu najprej definiramo različne namene (angl. intents), ki naj bi jih prepoznaval, nato posredujemo množico učnih primerov (povedi), ki so reprezentativni za posamezne namene. Agent nato z uporabo različnih pristopov strojnega učenja zgradi model, ki klasificira nov zahtevek uporabnika. Storitev DialogFlow omogoča tudi gradnjo modela, ki izlušči vrednosti predhodno definiranih parametrov iz teksta. Na primer, iz stavka: »Želim se naročiti pri zobozdravniku« bi sistem izluščil vrednost *zobozdravniku* kot iskani poseg. Gradnja modela poteka na podoben način. Storitvi posredujemo označene učne podatke na katerih nato sistem zgradi ustrezen model.

Agent vedno poskuša identificirati natančno storitev, ki naj bi zanimala uporabnika. V primeru, ko agent ni prepričan o posegu,

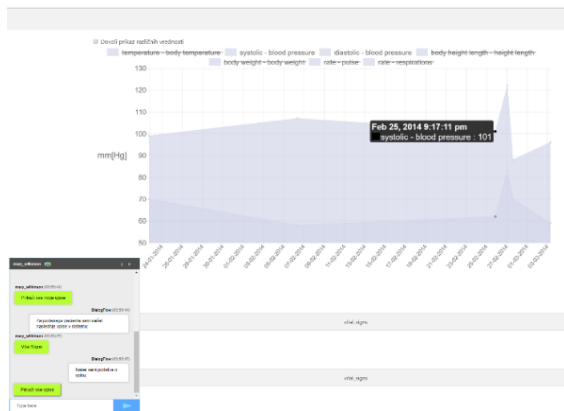
uporabniku vrne množico elementov, za katere je določil najvišjo verjetnost ujemanja. Ko sistem od uporabnika pridobi vse predvidene informacije, opravi zaledni sistem odločitev glede načina pridobitve želene informacije. V predstavljeni rešitvi se podatki pridobijo iz zunanjega vira (spletnega servisa). Zahtevo se nato pošlje do ustreznega vmesnika, ki je sposoben komunicirati z zunanjim virom. Ta pretvori JSON podatke v zahtevek, ki ga zunanja podatkovna zbirka, aplikacija ali platforma razume. Običajno se generira REST zahtevek. V primeru te aplikacije se pošlje zahtevek na API NIJZ [3].

Naslednja spletna storitev, kjer smo aplicirali metode jezikovnih tehnologij, je asistent za poizvedovanje po Think!EHR platformi. Pri implementaciji aplikacije smo ponovno uporabili generičnega svetovalca, kot je bil uporabljen v prejšnji aplikaciji, ki pa smo ga prilagodili glede na specifične domene.

Uporabnik se mora v aplikacijo najprej prijaviti, da se določi vloga: zdravnik ali pacient. Glede na določeno vlogo svetovalec nato posreduje različne tipe in obseg podatkov iz Think!EHR platforme. Zdravnik lahko pregleduje vse osebne podatke in meritve svojih pacientov. Uspešno prijavljena oseba (pacient) lahko pregleduje vse svoje zapise, ki so shranjeni v platformi Think!EHR. Asistent nato uporabniku posreduje celoten seznam zahtevanih podatkov, ki so zabeleženi za določeno osebo. Uporabnik nato s klikom na gumb izbere posamezno meritev.

Procesiranje tekstovnih zahtevkov je realizirano na enak način kot v prej predstavljeni rešitvi. Samo pridobivanje podatkov pa je realizirano preko REST vmesnika ehrscape platforme [5]. Gre za različico Think!EHR platforme, do katere je mogoče dostopati preko spleta. Določene podatke smo lahko pridobili preko osnovnih API klicev, za kompleksnejše poizvedbe pa smo uporabili AQL jezik, ki omogoča poizvedovanje po EHR zapisih.

Za pregledovanje meritev smo spletnemu vmesniku dodali grafični prikaz, ki se prikaže v ozadju pogovornega svetovalca (glej sliko 2). Ta omogoča prikaz poljubnih podatkov v meritvah na preglednejši način.



Slika 2: Grafični prikaz posamezne vrste meritev za vlogo pacient

3.4 Verige vrednosti

Verige vrednosti znotraj EMZ predvidevajo povezovanje različnih deležnikov in sodelovanje pri razvoju prototipov za različna področja, ki zahtevajo natančno specifično obravnavo. Dva primera tovrstne vzpostavitev verig vrednosti sta klinična obravnava bolnikov s kroničnimi boleznimi. Partnerji iz UKCLJ so definirali specifične klinične poti in intervence, ki so se nato integrirale v spletno platformo eOskrba (spletna platforma

projektne partnerja FRI). Druga oblika sodelovanja in primer vzpostavljene verige vrednosti je skupni razvoj treh partnerjev pri integraciji mobilnega EKG s platformo Think!EHR in zdravstvenim informacijskim sistemom. V ta namen so partnerji najprej definirali nov EHR arhetip za beleženje EKG meritev. Merilnik nato pošilja posamezne sklope meritev, v dogovorjenem formatu, v platformo Think!EHR. Zdravstveni portal pa iz platforme pridobiva podatke o meritvah za posameznega pacienta in zdravnikom omogoča pregleden prikaz v realnem času.

Za Slovenijo smo vzpostavili interesno skupino za področje EMZ, ki združuje preko 300 podjetij in institucij, ki delujejo ali pa so izkazala interes po sodelovanju na področju EMZ. Smiselna povezovanja preko interdisciplinarnih področij in verige vrednosti za področje zdravja smo podrobneje predstavili v Beli knjigi EMZ [6].

4. ZAKLJUČEK

V pričujočem prispevku smo opravili pregled projekta EMZ – Elektronsko in mobilno zdravstvo, ki se je izvajal v okviru programa Pametne specializacije S4 EkoSMART. Predstavili smo splošne koncepte in samo strukturo projekta ter fokusna področja na katere so se partnerji osredotočili.

Komunikacijska EMZ platforma omogoča komunikacijo v naravnem jeziku in ponuja intuitiven uporabniški vmesnik. Namenjena je splošni populaciji in njena glavna značilnost je enostavnost uporabe, kar bi lahko omogočilo tudi starejšim in manj večjim uporabe internetnih storitev dostop in uporabo tovrstnih naprednih storitev. Zajema pa tudi vrsto drugih zdravstvenih storitev in asistentov, recimo ASPO kot aplikacijo za odkrivanje spolnih infekcij ali 200 občinskih agentov, ki odgovarjajo na vprašanja o občinah. Glavne razvojne aktivnosti v projektu so bile usmerjene v razvoj prototipov IKT za področje zdravja. S strani partnerjev je bilo razvitih 21 različnih prototipov, ki naslavlajo posamezna področja. V prispevku predstavimo 9 primerov razvitih prototipov, ki kažejo večji tržni potencial ali uporabno vrednosti. Predstavljeni prototipi so iz različnih domen in rešujejo različne problematike starejših ali bolnikov.

Doseženi rezultati projekta bodo predstavljeni v Beli knjigi EMZ, ki bo predstavljena na posvetu v državnem svetu in predana odločevalcem na ministrstvih in v vladi.

5. ZAHVALA

Raziskave in razvoj so nastale v okviru programa EkoSMART in so delno sofinancirane s strani Ministrstva za izobraževanje, znanost in šport in Evropske unije iz Evropskega sklada za regionalni razvoj (ESRR).

6. REFERENCE

- [1] <http://ekosmart.net/sl/ekosmart/>, dostop, 28. 9. 2019
- [2] <https://dialogflow.com/>, dostop, 28. 9. 2019
- [3] <https://cakalnedobe.ezdrav.si/>, dostop 28. 9. 2019
- [4] <https://www.ehealthobjects.com/our-solutions/thinkehr/>, dostop 29. 9. 2019
- [5] <https://www.ehrscape.com/api-explorer.html>, dostop 29. 9. 2019
- [6] Bela knjiga EMZ, <http://library.ijs.si/Stacks/Literature/Bela%20knjiga%20EMZ%20EkoSMART.pdf>
- [7] <https://www.infonet.si/produkti/birpis21/>, dostop 24. 9. 2019
- [8] <https://rocket.chat/>, dostop 29. 9. 2019

Five Attempts at Cross-Dataset Speech Emotion Recognition

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

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

ABSTRACT

In this paper, we tried to recognize emotions from speech when we have a very small dataset available. Because recording and annotating new data is a costly task, our goal is to use publicly available datasets in addition to our own to improve the recognition accuracy. To do this, we implemented five different methods able to extract knowledge from the publicly available dataset and use it in our target dataset. Two of these methods are based on transfer learning, one is a multi-task learning method, and the last two use advanced feature normalization methods to bring the feature domains from the two datasets closer together. We show that in certain combinations of train-test set, some of our methods outperform the baseline classifier by a maximum of 9 percentage points. In some cases, however, the baseline method proved to provide best results.

Keywords

speech emotion recognition, transfer learning, multi-task learning, cross-dataset

1. INTRODUCTION

In the last two decades a lot of research has been put into the automatic recognition of emotional states. The main reason for this is the rapid development of affective user interfaces. If we are able to recognize the user's emotions, we can develop dynamic applications that can adapt to the user's feelings at any given time. Here, our task could be to measure the stress of the calling users and include them in a priority list accordingly.

In our research, we focused on measuring the emotional state of a person based on their speech. More precisely, we did not analyse what the person is saying, but how they express themselves. Since this type of emotion recognition does not analyse the content of the person's speech and does not require any of their other personal data, it can be used if we have some limitations regarding the use of personal data. The main obstacle to determining a person's emotional state for machine learning is the lack of data. Bringing people into a certain emotional state is a challenging and, when it comes to negative emotions, unethical task. Furthermore, recording people expressing their genuine feelings, without their knowledge, violates the right to privacy, and these datasets, if ever acquired, are not publicly available. Therefore, most of the datasets we get are from actors who do their best to perform certain emotions.

Many papers present high accuracy scores when training

and testing a model with the same dataset [9, 6, 10] but, what happens if we do not have data for our specific problem? What happens if we use recording devices of different quality, or if our target subjects are older people and not young or middle-aged actors, or if our setting differs in some other way? In our research, we assumed that we have a small amount of representative data of our problem, and we wanted to additionally exploit publicly available datasets. The most similar related work is a contribution by Latif et al. from INTERSPEECH 2018 [8]. Although it claims to achieve state-of-the-art results, when trained on a source dataset and tested on a target its accuracy does not even exceed the majority classifier. When a small amount of data from the target dataset was used, a modest improvement was observed, but the paper only distinguishes between two emotions. It thus seems that the issue we are tackling in this paper is poorly explored in the literature.

The most common way to use information from a source dataset to improve the classifier of a target dataset is to use transfer learning. We have chosen two different methods for transfer learning. The first method was recreated from the paper mentioned previously, while the other method uses Fully Connected Neural Networks to transfer some of network parameters between datasets. Additionally, we tried multi-task learning as well as two different types of feature normalization, which we applied on the data in order to bring the feature domains between the two datasets closer together.

In Section 2 we present the datasets we use. In Section 3 we present the five methods we used for cross-dataset emotion recognition. In Section 4 we present our evaluation methods and the achieved results. Finally in Section 5, we conclude and present our future work.

2. DATASETS

To detect emotions from speech, we used four publicly available datasets: EmoDB [1], EMOVO [3], IEMOCAP [2] and SAVEE [7]. The datasets were recorded in three different languages: IEMOCAP and SAVEE datasets were recorded in English, while EMOVO and EmoDB were recorded in Italian and German. A common problem when combining multiple datasets is that most of the datasets use different sets of emotions. To deal with this problem, we used only instances presenting four basic types of emotions, which are present in all the datasets: neutral, anger, joy and sadness. The number of instances for each emotion per dataset is presented in Table 1.

Table 1: Number of instances per emotion

Dataset	Neutral	Anger	Joy	Sadness
EmoDB	79	127	71	62
EMOVO	84	84	84	84
IEMOCAP	392	500	94	467
SAVEE	120	60	60	60

3. METHODS

The best established way to build a model able to recognize emotions from speech is by extracting global features from the speech and then building a classifier on top of these features. To extract features, we used a publicly available toolkit - OpenSmile [4], which offers a wide range of possible sets of features. We decided to use the ‘emobase2010’ feature set. This set is composed of overall 1582 features. As the machine learning algorithm we selected Random Forest with 1000 trees and maximal depth of 10. This combination outperformed several alternatives, including Deep Learning on raw audio. An additional advantage of the OpenSmile features and Random Forest over Deep Learning is that features can easily be extracted on the phone, so that raw audio is never sent outside the user’s device. We developed or implemented five methods for transfer learning, which we describe in sections 3.1-3.5.

3.1 Deep Belief Network

In speech emotion recognition, there has not been much related work whose main focus is to transfer knowledge from the source dataset to the target dataset. The most dedicated attempt is the already mentioned one by Latif et al. [8], in which they used Deep Belief Network (DBN). To evaluate their method, they used another work that used autoencoders to transfer knowledge from the source dataset to the target dataset. They achieved better results than the autoencoders approach, and therefore we decided to recreate their method. In our DBN implementation, we used the same network parameters as described in their paper.

3.2 Fully Connected Deep Neural Network

Since the rise of transfer learning, the most commonly used method of transferring knowledge from one problem in another is by transferring network parameters. Therefore, in the second method we trained a Fully Connected Neural Network (FCNN) on the source dataset and transferred some of the network parameters to the target dataset. The FCNN architecture is composed of one input layer, one output layer, and three hidden layers. The input layer takes the same amount of input units as the number of features extracted from one utterance. The first hidden layer is composed of 1000 units, the second hidden layer is composed of 500 units and the third hidden layer is composed of 300 units. The output layer consists of only four units, one for each emotion. The activation function of all layers is ‘tanh’. The only exception is the output layer, which uses ‘softmax’ activation function.

First, the FCNN was trained on the whole source dataset. After the training on the source dataset was finished, and all network parameters have been determined, we froze all parameters of the network, except those belonging to the output layer. We then fine-tuned the parameters of the final layer using a part of the instances from the target dataset.

3.3 Multi-task Learning

In the multi-task learning method, we used the same Random Forest classifier as the one described in the baseline method. However, instead of having the same target class for matching emotions from the source and the target dataset, we used two different target classes: one for the emotion from the source dataset, and another one for the same emotion from the target dataset. For example, angry utterances from the source and the target dataset would get the same target label ‘Anger’ in the baseline Random Forest Classifier. However, in the multi-task learning approach, angry utterances from the source dataset would get the target label ‘Anger1’, while angry utterances from the target dataset would get the target label ‘Anger2’. The idea is that the classifier classifies specifically into classes of the target dataset, while the structure of the classifier still benefits from the source dataset (upper leaves of the tree in Random Forest). Because when training we used the whole source dataset and only a small portion from the target dataset, we ended up with unequal distribution of emotions. To deal with this problem, we oversampled examples from the target dataset until we got equal distributions in both datasets.

3.4 Normalization based on neutral speech

As shown in Table 1, the distribution of emotions is not equal across dataset and thus, a simple feature normalization and standardization method might not work across different datasets.

To implement a more advanced feature normalization method, we applied a normalization technique on the source and on the target data independently. In this normalization technique, we used neutral speech to bring the datasets to the same reference point. Ideally the neutral speech using our normalization technique should be near the coordinate space origin. To normalize and standardize our data, we applied the following formula to the feature values:

$$x_{i_new} = \frac{x_i - \mu}{\sigma}$$

where μ is the average value from neutral speech in the training data, σ is the standard deviation in all training data and x_i is the i -th instance in the data.

To evaluate the performance of the model whose features were normalized based on the neutral emotion, we used the baseline Random Forest Classifier.

3.5 Normalization based on feature distribution

When analysing feature distributions between two datasets, we noticed that most of the features do not have the same distributions per emotion. For example, on the left side in Figure 1 we present the distribution of feature ‘pcm_loudness_sma_amean’ on neutral utterances in IEMOCAP, while on the right side, the distribution of the same feature is presented for neutral utterances in SAVEE. Since this could be confusing for our model, we tried to bring the two feature distributions as close as possible for each emotion.

To do this, we used the feature distribution from the training data in the target dataset as the baseline, and tried to bring the feature distribution from the whole source dataset as close as possible to the baseline distribution. Thus, for each emotion, we divided the feature distribution of the training data in the target dataset into 5 equal bins and saw how

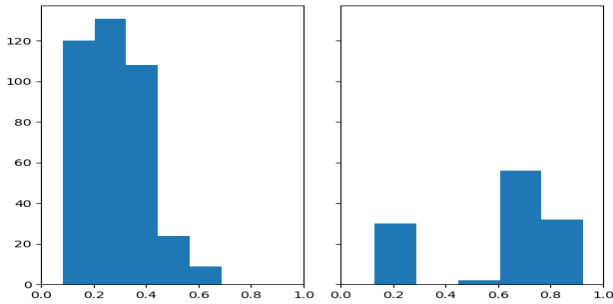


Figure 1: Distribution of values in neutral speech from feature `pcm_loudness_sma_amean` on the left side in IEMOCAP, on the right side for SAVEE

much of the data belongs to each bin. Thus, all instances in the target dataset whose $feature_i$ belong to the first bin of the $feature_i$ distribution, could be given the value 0, all instances that belong to the second bin of the $feature_i$ distribution would be given the value 1, etc.

Let us assume that the percentage of the training data from the target dataset that was given the value 0 for $feature_i$ is x_0 , the percentage of the training data that was given the value 1 is x_1 , etc. To bring the distributions for $feature_i$ from the source and the target dataset closer together, we would assign the lowest x_0 percent of data from the source dataset a value of 0, etc. Thus, we got a similar distribution of the features in the source and the target dataset for each emotion separately.

To evaluate the performance of the model whose features were normalized based on their distribution, we used the baseline Random Forest Classifier.

4. EVALUATION AND RESULTS

Recording and annotating data is an expensive process. Because of this, for training our models we used at most one subject, while the rest of the subjects would be used for testing the performance of the model. To evaluate the performance of the model on the target dataset without a transfer-learning method, we used three different scenarios:

- In the first scenario, we use the whole source dataset for training, and the whole target dataset for testing. This simulates the case when we do not have any of our target data, thus only training our model on a publicly available dataset.
- In the second scenario, we used one subject from our target dataset as the training set and the rest of the subjects as the testing set. To get a more objective evaluation of the performance of the model, each of the subjects was used as training data once, and the final result was calculated by averaging the accuracy of each train-test split. This scenario simulates the case where we only use our small dataset for training.
- In the third scenario we used the whole source dataset and one subject from the target dataset for training. The rest of the subjects from the target dataset were used for testing. Similarly as in the second scenario, each subject from the target dataset was used as the

training data once, making the final result an average of each train-test split. This scenario simulates the case where we combine our small dataset with some publicly available dataset.

Since only two of the four datasets we use are recorded in the same language, we decided to evaluate the models only on these two datasets. This way, we can compare same-language and different-language cross-dataset emotion recognition on the same target datasets. To present the complexity of the task, in Table 2 we used the Majority Classifier and Random Forest Classifier for each of the three possible scenarios. The results achieved using the first scenario are the poorest, not even achieving the majority classifier. The results achieved using the second and the third scenario outperform the majority classifier, but still the classifier from the third scenario gives overall poorer results compared to the classifier from the second scenario. This could mean that we do not gain any useful information from the source dataset.

We applied the transfer learning methods from Section 3 according to the third scenario. The results are presented in Table 3. To evaluate the success of the information transfer, we compared these results to the baseline Random Forest Classifier calculated using the second scenario. The results in Table 3 show us that most of the improvements are achieved by normalizing the feature spaces based on the feature distribution. However, the presented results are not optimistic, since in some cases the best results were achieved using the baseline classifier. So far, the best improvement we achieved was 9 percentage points, which we gained when training on EmoDB and testing on SAVEE while normalizing the features based on their distributions. This method outperformed both the DBN presented as most suitable for this type of problems in related work, as well as the commonly used FCNN transfer learning.

An interesting observation is that when our methods use EmoDB and EMOVO as train data and SAVEE as test data, they perform better compared to when the same-language IEMOCAP is used as train data. This happened with most of our methods, and could indicate that the way the recording took place (5 min conversations vs. short utterances), might be more important when choosing which source dataset to use, than the language.

5. CONCLUSIONS

In this paper, we tried to use the knowledge obtained from a source dataset in order to improve the classification accuracy of a target dataset. We found that although in different languages, EmoDB and EMOVO contain more useful information for detecting emotions from speech in SAVEE, compared to the same-language database IEMOCAP.

The baseline classifier could be outperformed by using some of the methods described here, with a maximum improvement of 9 percentage points. The best performance was achieved by normalizing the features, based on their distributions. The worst performance was achieved by a method from related work, which did not even outperform the majority classifier.

Although we implemented five different methods for cross-dataset speech emotion recognition, there are other possibilities. A potentially more effective, but substantially more

Table 2: Results obtained from the majority classifier and baseline Random Forest Classifier for each scenario without transfer learning

Train dataset	Test dataset	Majority	Scenario1	Scenario2	Scenario3
EmoDB	SAVEE	40%	29%	49%	57%
EMOVO	SAVEE	40%	41%	49%	51%
IEMOCAP	SAVEE	40%	27%	49%	41%
EmoDB	IEMOCAP	34%	34%	67%	62%
EMOVO	IEMOCAP	34%	52%	67%	67%
SAVEE	IEMOCAP	34%	33%	67%	65%

Table 3: Results obtained from the majority classifier and baseline Random Forest Classifier compared to the five transfer learning methods

Train dataset	Test dataset	Majority	Baseline RF	DBN	FCNN	Multi-task	Norm.1	Norm.2
EmoDB	SAVEE	40%	49%	20%	50%	56%	57%	58%
EMOVO	SAVEE	40%	49%	30%	58%	50%	51%	58%
IEMOCAP	SAVEE	40%	49%	20%	46%	42%	41%	47%
EmoDB	IEMOCAP	34%	67%	33%	60%	62%	62%	66%
EMOVO	IEMOCAP	34%	67%	30%	61%	67%	67%	67%
SAVEE	IEMOCAP	34%	67%	27%	56%	66%	65%	69%

complex approach may be by using Generative Adversarial Networks [5] to translate the features from one dataset to another. This is the main approach we consider for future work, assuming it can work on modestly sized dataset such as are available.

A cursory look at the literature suggests that emotion recognition from speech is not a very difficult problem, since many papers report good results and several datasets are publicly available. However, our study shows that practical applicability of these datasets is limited considering how poorly cross-dataset learning works. It is also possible that the typical methods for emotion recognition from speech would prove unsuitable for the wider range of emotion expressed in real life. Therefore, it is important to study emotion recognition without limiting to one homogeneous dataset.

6. ACKNOWLEDGEMENTS

We would like to thank our colleagues Benjamin Fele and Aljoša Vodopija for their help in this research. This Research has received funding from the European Union’s Horizon 2020 research and innovation programme under Grant Agreement No 769765

7. REFERENCES

- [1] F. Burkhardt, A. Paeschke, M. Rolfes, W. F. Sendlmeier, and B. Weiss. A database of german emotional speech. In *Ninth European Conference on Speech Communication and Technology*, 2005.
- [2] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, and S. S. Narayanan. Iemocap: Interactive emotional dyadic motion capture database. *Language resources and evaluation*, 42(4):335, 2008.
- [3] G. Costantini, I. Iaderola, A. Paoloni, and M. Todisco. Emovo corpus: an italian emotional speech database. In *International Conference on Language Resources and Evaluation (LREC 2014)*, pages 3501–3504. European Language Resources Association (ELRA), 2014.
- [4] F. Eyben, M. Wöllmer, and B. Schuller. Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1459–1462. ACM, 2010.
- [5] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.
- [6] K. Han, D. Yu, and I. Tashev. Speech emotion recognition using deep neural network and extreme learning machine. In *Fifteenth annual conference of the international speech communication association*, 2014.
- [7] S. Haq, P. J. Jackson, and J. Edge. Audio-visual feature selection and reduction for emotion classification. In *Proc. Int. Conf. on Auditory-Visual Speech Processing (AVSP’08), Tangalooma, Australia*, 2008.
- [8] S. Latif, R. Rana, S. Younis, J. Qadir, and J. Epps. Transfer learning for improving speech emotion classification accuracy. *arXiv preprint arXiv:1801.06353*, 2018.
- [9] T. L. Nwe, S. W. Foo, and L. C. De Silva. Speech emotion recognition using hidden markov models. *Speech communication*, 41(4):603–623, 2003.
- [10] G. Trigeorgis, F. Ringeval, R. Brueckner, E. Marchi, M. A. Nicolaou, B. Schuller, and S. Zafeiriou. Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network. In *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5200–5204. IEEE, 2016.

Indeks avtorjev / Author index

Andova Andrejaana.....	81
Bašić Ivica.....	5
Bizjak Jani.....	77
Bohanec Marko.....	5
Bosnić Zoran.....	41
Brockhoff Dimo.....	37
Budna Borut.....	9, 25
Čebokli Edi.....	61
Cigale Matej.....	29
De Masi Carlo M.....	13
Debelak Klemen.....	5
Dovgan Erik.....	45
Fele Benjamin.....	49
Filipič Bogdan.....	65
Gams Matjaž.....	17, 21, 53, 69, 77
Georgiev Dejan.....	61
Gjoreski Hristijan.....	73
Gjoreski Martin.....	9, 69
Golob David.....	17
Gradišek Anton.....	9
Grasselli Gregor.....	21, 53
Hansen Nikolaus.....	37
Janko Vito.....	49
Katrašnik Marko.....	57
Kocuvan Primož.....	21
Kolenik Tine.....	69
Lukan Junoš.....	57
Luštrek Mitja.....	13, 25, 29, 45, 49, 57, 81
Markovska Marija.....	73
Mlakar Miha.....	25
Papič Aleš.....	41
Poglajen Gregor.....	9
Reščič Nina.....	49
Romih Miro.....	33
Šef Tomaž.....	33
Slapničar Gašper.....	29
Štrubelj Luka.....	5
Štruc Vitomir.....	57
Tashkovski Dimitar.....	73
Tavčar Aleš.....	17, 21, 77
Tušar Tea.....	37
Valič Jakob.....	25
Velichkovska Bojana.....	73
Vodopija Aljoša.....	65
Volz Vanessa.....	37
Vrbanić Ivan.....	5
Žabkar Jure.....	61
Žganec Gros Jerneja.....	33
Zupančič Jernej.....	53

IS
20
19

Konferenca / Conference

Uredili / Edited by

**Slovenska konferenca o umetni inteligenci /
Slovenian Conference on Artificial Intelligence**

Mitja Luštrek, Matjaž Gams, Rok Piltaver