Zbornik 28. mednarodne multikonference

INFORMACIJSKA DRUŽBA – IS 2025

Zvezek A

Proceedings of the 28th International Multiconference

INFORMATION SOCIETY – IS 2025

Volume A

Slovenska konferenca o umetni inteligenci
Slovenian Conference on Artificial Intelligence

Uredniki / Editors

Mitja Luštrek, Matjaž Gams, Rok Piltaver

http://is.ijs.si

8.–9. oktober 2025 / 8–9 October 2025
Ljubljana, Slovenia

Uredniki:

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

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

Rok Piltaver
Outfit7, Ljubljana

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

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

Ljubljana, oktober 2025

Informacijska družba
ISSN 2630-371X

PREDGOVOR MULTIKONFERENCI INFORMACIJSKA DRUŽBA 2025

28. mednarodna multikonferenca *Informacijska družba* se odvija v času izjemne rasti umetne inteligence, njenih aplikacij in vplivov na človeštvo. Vsako leto vstopamo v novo dobo, v kateri generativna umetna inteligenca ter drugi inovativni pristopi oblikujejo poti k superinteligenci in singularnosti, ki bosta krojili prihodnost človeške civilizacije. Naša konferenca je tako hkrati tradicionalna znanstvena in akademsko odprta, pa tudi inkubator novih, pogumnih idej in pogledov.

Letošnja konferenca poleg umetne inteligence vključuje tudi razprave o perečih temah današnjega časa: ohranjanje okolja, demografski izzivi, zdravstvo in preobrazba družbenih struktur. Razvoj UI ponuja rešitve za številne sodobne izzive, kar poudarja pomen sodelovanja med raziskovalci, strokovnjaki in odločevalci pri oblikovanju trajnostnih strategij. Zavedamo se, da živimo v obdobju velikih sprememb, kjer je ključno, da z inovativnimi pristopi in poglobljenim znanjem ustvarimo informacijsko družbo, ki bo varna, vključujoča in trajnostna.

V okviru multikonference smo letos združili dvanajst vsebinsko raznolikih srečanj, ki odražajo širino in globino informacijskih ved: od umetne inteligence v zdravstvu, demografskih in družinskih analiz, digitalne preobrazbe zdravstvene nege ter digitalne vključenosti v informacijski družbi, do raziskav na področju kognitivne znanosti, zdrave dolgoživosti ter vzgoje in izobraževanja v informacijski družbi. Pridružujejo se konference o legendah računalništva in informatike, prenosu tehnologij, mitih in resnicah o varovanju okolja, odkrivanju znanja in podatkovnih skladiščih ter seveda Slovenska konferenca o umetni inteligenci.

Poleg referatov bodo okrogle mize in delavnice omogočile poglobljeno izmenjavo mnenj, ki bo pomembno prispevala k oblikovanju prihodnje informacijske družbe. »Legende računalništva in informatike« predstavljajo domači »Hall of Fame« za izjemne posameznike s tega področja. Še naprej bomo spodbujali raziskovanje in razvoj, odličnost in sodelovanje; razširjeni referati bodo objavljeni v reviji *Informatica,* s podporo dolgoletne tradicije in v sodelovanju z akademskimi institucijami ter strokovnimi združenji, kot so ACM Slovenija, SLAIS, Slovensko društvo Informatika in Inženirska akademija Slovenije.

Vsako leto izberemo najbolj izstopajoče dosežke. Letos je nagrado *Michie-Turing* za izjemen življenjski prispevek k razvoju in promociji informacijske družbe prejel **Niko Schlamberger**, priznanje za raziskovalni dosežek leta pa **Tome Eftimov**. »Informacijsko limono« za najmanj primerno informacijsko tematiko je prejela odsotnost obveznega pouka računalništva v osnovnih šolah. »Informacijsko jagodo« za najboljši sistem ali storitev v letih 2024/2025 pa so prejeli Marko Robnik Šikonja, Damir Vreš in Simon Krek s skupino za slovenski veliki jezikovni model GAMS. Iskrene čestitke vsem nagrajencem!

Naša vizija ostaja jasna: prepoznati, izkoristiti in oblikovati priložnosti, ki jih prinaša digitalna preobrazba, ter ustvariti informacijsko družbo, ki koristi vsem njenim članom. Vsem sodelujočim se zahvaljujemo za njihov prispevek — veseli nas, da bomo skupaj oblikovali prihodnje dosežke, ki jih bo soustvarjala ta konferenca.

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

FOREWORD TO THE MULTICONFERENCE INFORMATION SOCIETY 2025

The 28th International Multiconference on the Information Society takes place at a time of remarkable growth in artificial intelligence, its applications, and its impact on humanity. Each year we enter a new era in which generative AI and other innovative approaches shape the path toward superintelligence and singularity — phenomena that will shape the future of human civilization. The conference is both a traditional scientific forum and an academically open incubator for new, bold ideas and perspectives.

In addition to artificial intelligence, this year's conference addresses other pressing issues of our time: environmental preservation, demographic challenges, healthcare, and the transformation of social structures. The rapid development of AI offers potential solutions to many of today's challenges and highlights the importance of collaboration among researchers, experts, and policymakers in designing sustainable strategies. We are acutely aware that we live in an era of profound change, where innovative approaches and deep knowledge are essential to creating an information society that is safe, inclusive, and sustainable.

This year's multiconference brings together twelve thematically diverse meetings reflecting the breadth and depth of the information sciences: from artificial intelligence in healthcare, demographic and family studies, and the digital transformation of nursing and digital inclusion, to research in cognitive science, healthy longevity, and education in the information society. Additional conferences include Legends of Computing and Informatics, Technology Transfer, Myths and Truths of Environmental Protection, Knowledge Discovery and Data Warehouses, and, of course, the Slovenian Conference on Artificial Intelligence.

Alongside scientific papers, round tables and workshops will provide opportunities for in-depth exchanges of views, making an important contribution to shaping the future information society. *Legends of Computing and Informatics* serves as a national »Hall of Fame« honoring outstanding individuals in the field. We will continue to promote research and development, excellence, and collaboration. Extended papers will be published in the journal *Informatica*, supported by a long-standing tradition and in cooperation with academic institutions and professional associations such as ACM Slovenia, SLAIS, the Slovenian Society Informatika, and the Slovenian Academy of Engineering.

Each year we recognize the most distinguished achievements. In 2025, the Michie-Turing Award for lifetime contribution to the development and promotion of the information society was awarded to **Niko Schlamberger**, while the Award for Research Achievement of the Year went to **Tome Eftimov**. The »Information Lemon« for the least appropriate information-related topic was awarded to the absence of compulsory computer science education in primary schools. The »Information Strawberry« for the best system or service in 2024/2025 was awarded to Marko Robnik Šikonja, Damir Vreš and Simon Krek together with their team, for developing the Slovenian large language model GAMS. We extend our warmest congratulations to all awardees.

Our vision remains clear: to identify, seize, and shape the opportunities offered by digital transformation, and to create an information society that benefits all its members. We sincerely thank all participants for their contributions and look forward to jointly shaping the future achievements that this conference will help bring about.

Mojca Ciglarič, Chair of the Program Committee
Matjaž Gams, Chair of the Organizing Committee

KONFERENČNI ODBORI
CONFERENCE COMMITTEES

International Programme Committee

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

Organizing Committee

Matjaž Gams, chair
Mitja Luštrek
Lana Zemljak
Vesna Koricki
Mitja Lasič
Blaž Mahnič

Programme Committee

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

KAZALO / TABLE OF CONTENTS

Zbornik 28. mednarodne multikonference

INFORMACIJSKA DRUŽBA – IS 2025

Zvezek A

Proceedings of the 28th International Multiconference

INFORMATION SOCIETY – IS 2025

Volume A

Slovenska konferenca o umetni inteligenci
Slovenian Conference on Artificial Intelligence

Uredniki / Editors

Mitja Luštrek, Matjaž Gams, Rok Piltaver

http://is.ijs.si

8.–9. oktober 2025 / 8–9 October 2025
Ljubljana, Slovenia

PREDGOVOR SLOVENSKI KONFERENCI O OMETNI INTELIGENCI

Slovenska konferenca o umetni inteligenci se letos odvija v času, ko umetna inteligenca še naprej intenzivno prodira v znanost, industrijo in vsakdanje življenje, še nikoli tako hitro in tako koristno. Še vedno so v ospredju veliki jezikovni modeli, ki so svoje zmožnosti razumevanja in generiranja že uspešno razširili na zvok, slike in video. Zanimivo raziskovalno področje so tudi temeljni (angl. foundation) modeli za druge vrste podatkov – npr. senzorskih in bioloških, pa tudi takih za robotske akcije, ki jih je takisto mogoče povezati z jezikom. Ti modeli so posebej dragoceni v medicinskih raziskavah, kjer so že privedli do razvoja novih zdravilnih učinkovin. Tovrstne raziskave bodo morda privedle do modelov, ki bodo znali celostno razumevati svet in nanj tudi vplivati, kar močno diši po umetni splošni inteligenci.

Najnaprednejše raziskave umetne inteligence danes zahtevajo infrastrukturo, ki je v Sloveniji nimamo in se je tudi ne moremo nadejati, vseeno pa se je v zadnjem letu tudi v domačih logih zgodilo marsikaj zanimivega. Največji dogodek je bil bržkone pridobitev financiranja za Slovensko tovarno umetne inteligence – superračunalnik za 150 milijonov EUR, prilagojen umetni inteligenci. Poleg tega je bil zgrajen velik jezikovni model za slovenščino GaMS, ki omogoča boljše izražanje v našem jeziku in prispeva k slovenski digitalni suverenosti. V Sloveniji nastaja tudi veliko aplikacij, ki uporabljajo velike jezikovne modele. Med njimi bi radi izpostavili zdravstvenega pomočnika HomeDOCtor, ki zna državljanom svetovati glede zdravstvenih težav bolje kot splošnonamenski modeli.

Vrnimo se zdaj h konferenci: letos ima 21 prispevkov, kar je največ po rekordnem letu 2020. Od teh jih dve tretjini prihajata z Instituta Jožef Stefan, kar ne odstopa dosti od statistike zadnjih let. Tako širša zastopanost različnih slovenskih ustanov vključno z industrijo še vedno ostaja naša želja. Ponosni smo, da smo letošnjo konferenco obogatili s kar tremi posebnimi dogodki. Otvoritev sestavljata uvodni nagovor predstavnice Ministrstva za digitalno preobrazbo in vabljeno predavanje Eve Tube, ki je v Slovenijo prišla na prestižno mesto ERA Chair v okviru projekta AutoLearn-SI. Ker umetna inteligenca prodira v vse pore našega življenja, med katere sodi tudi umetnost, smo zato organizirali sekcijo Beyond Human Art prav na to temo. In nenazadnje smo Slovensko tovarno umetne inteligence obeležili s sekcijo, kjer smo se poučili o tovarni, njeni uporabi v znanstvenih raziskavah in vlogi pri obdelavi senzorskih podatkov.

Konferenca ostaja enkraten slovenski in mednarodni prostor odličnosti, odprte akademske razprave in novih idej. Ponosni smo, da skupaj gradimo slovensko skupnost umetne inteligence, ki s svojim znanjem in inovacijami prispeva k reševanju ključnih izzivov sodobnega časa ter krepi vlogo Slovenije v evropskem in svetovnem prostoru.

Mitja Luštrek
Matjaž Gams
Rok Piltaver

FOREWORD TO SLOVENIAN CONFERENCE ON ARTIFICIAL INTELLIGENCE

Slovenian Conference on Artificial Intelligence is taking place this year at a time when AI continues to advance rapidly into science, industry, and everyday life, faster and more usefully than ever before. Large language models are still at the forefront, having already successfully expanded their capabilities to the understanding and generation of sound, images and video. An interesting research area includes foundation models for other types of data – for example, sensor and biological data, as well robotic actions, which can likewise be connected to language. These models are especially valuable in medical research, where they have already led to the development of new therapeutic compounds. Such research may eventually result in models capable of comprehensively understanding the world and interacting with it, which strongly suggests artificial general intelligence.

The most advanced artificial intelligence research today requires infrastructure that Slovenia does not have and cannot realistically expect, yet the past year has nevertheless seen several significant and interesting advances in Slovenia as well. The most important milestone was probably securing the funding for the Slovenian Artificial Intelligence Factory – a 150 million EUR supercomputer tailored to artificial intelligence. In addition, a large language model for Slovenian, GaMS, was built, enabling better expression in our language and contributing to Slovenian digital sovereignty. Slovenia is also seeing the rise of many applications that make use of large language models. Among them we would like to highlight the healthcare assistant HomeDOCtor, which is able to advise citizens on health issues better than general-purpose models.

Returning to the conference: this year, it features 21 papers, the highest number since the record year of 2020. Out of these, two thirds come from Jožef Stefan Institute, which does not differ much from the statistics of recent years. Thus, a broader representation of various Slovenian institutions, including industry, remains our goal. We are proud that this year's conference was enriched with three special events. The opening included a welcome address by a representative of the Ministry of Digital Transformation and a keynote lecture by Eva Tuba, who came to Slovenia to take up the prestigious ERA Chair position within the AutoLearn-SI project. Since artificial intelligence is making its way into every aspect of our lives, including art, we organized a special section titled Beyond Human Art dedicated to this theme. Finally, we marked the Slovenian Artificial Intelligence Factory with a session where we learned about the factory itself, its use in scientific research, and its role in processing sensor data.

The conference is a unique Slovenian and international venue for excellence, open academic debate and new ideas. We are proud that together we are building the Slovenian AI community, which, through its knowledge and innovations, contributes to addressing the key challenges of our time and strengthens Slovenia's role in Europe and globally.

Mitja Luštrek
Matjaž Gams
Rok Piltaver

PROGRAMSKI ODBOR / PROGRAMME COMMITTEE

Detecting Pollinators from Stem Vibrations Using a Neural Network

Žan Ambrožič
za44564@student.uni-lj.si
Faculty of Mathematics and Physics, University of Ljubljana
Ljubljana, Slovenia

Lorenzo Bianco
l.bianco@unito.it
Department of Life Science and System Biology, University of Turin
Turin, Italy

Rok Šturm
rok.sturm@nib.si
National Institute for Biology
Ljubljana, Slovenia

David Susič, Maj Smerkol, Anton Gradišek
Department of Intelligent Systems, Jožef Stefan Institute
Ljubljana, Slovenia

Abstract

Passive sensing of pollinator activity is important for biodiversity monitoring and conservation, yet conventional acoustic or visual methods produce large amounts of data and face deployment challenges. In this work, we present initial results on investigating stem vibration as an alternative signal for detecting pollinator presence on flowers. Vibration recordings were collected with a laser vibration instrument from various flower species at multiple locations in Slovenia, totaling approximately 14 hours, of which 3 hours were expert-annotated as positive (insect activity present). The task was formulated as a binary classification problem: determining whether a vibration segment corresponds to a pollinator physically touching the flower. Using a neural network model, performance was evaluated with five-fold cross-validation across three experiments: (i) using a balanced subset, (ii) using the full dataset, and (iii) using the full dataset with heuristic prediction smoothing. On the balanced subset, the model achieved an average F1-score of 0.86 ± 0.06; on the full dataset, 0.62 ± 0.07; and with heuristic smoothing, 0.69 ± 0.11, demonstrating both the feasibility of vibration-based detection and the benefits of post-processing. Beyond binary detection, future work will focus on species- and activity-level classification. Ultimately, the goal is to develop lightweight vibration detectors deployable directly on plants, enabling scalable estimation of pollinator visitation rates in natural and agricultural environments.

Keywords

stem vibrations, pollination, neural networks, buzz detection, spectrograms

1 Introduction

Europe supports a rich diversity of wild pollinators among them 2,051 species of bees and 892 species of hoverflies. Collectively, pollinators provide a wide range of benefits to society including more than €15 billion per year contribution to the market value of European crops, pollinating around 78 percent of wild flowering plants. This pollination service ensures healthy ecosystem functioning and maintains wider biodiversity as well as culturally important flower-rich landscapes [1]. Many reviews highlight the global decline in insects [2], [3] and in particular wild bees [4], [5]. Internationally, the UN Intergovernmental science-policy Platform on Biodiversity and Ecosystem Services (IPBES) and the Convention on Biological Diversity (CBD) highlighted the need to collect long-term high-quality data on pollinators and pollination services in order to direct policy and practice responses to address the decline. There were already some attempts to monitor pollinators' activity from sound/soundscapes recordings (e.g. [6]). Here, we explore for the first time to monitor pollination activity by using vibroscape recordings [7] from flowering plants which are visited by different pollinators. We investigated the possibility of neural networks for automatic detection of pollinator visits on flowers.

2 Dataset

The dataset comprises vibration waveforms from flowers, which were used for model training, and auxiliary audio and camera recordings collected for labeling and species identification. All recordings were obtained during July and August 2024 at various locations in Slovenia. The vibrations were measured using a VibroGo (Polytec, Germany) laser vibration instrument, which has an operational range of up to 30 m and can detect movements up to 6 $\mathrm{m\,s^{-1}}$ at frequencies up to 320 kHz. For this study, measurements were performed at close range, with a frequency span of 0–24 kHz and a sampling rate of 48 kHz.

For the measurements, the flower stem was fixed to a pole to minimize large movements, and a small piece of reflective foil was attached slightly below the flower to enable the laser vibrometer to capture fine vibrations caused by insect activity. Our data acquisition setup is shown in Figure 1.

The dataset comprised vibration recordings of up to 10 minutes each, collected from flowers belonging to the genera *Calystegia*, *Cichorium* (the majority of samples), *Crepis*, *Epilobium*, *Knautia*, *Leontodon*, *Lotus*, *Pastinaca*, and *Trifolium*. In total, the recordings amounted to approximately 14 hours, of which 3 hours were annotated for insect activity (as positive), while the rest did not contain insect activity and was considered negative. Labeling was performed in Raven Pro by expert annotators, who used synchronized audio and video recordings to confirm insect presence and identify species. Each recording was annotated with time intervals indicating insect activity, insect species, activity type, and, when relevant, additional notes. For the purpose of this study, where we are only interested in binary classification of detecting pollinators, all intervals with any insect activity which included physically touching the flower were labeled as 1, and 0 otherwise.

Information Society 2025, Ljubljana, Slovenia

https://doi.org/10.70314/is.2025.skui.5707

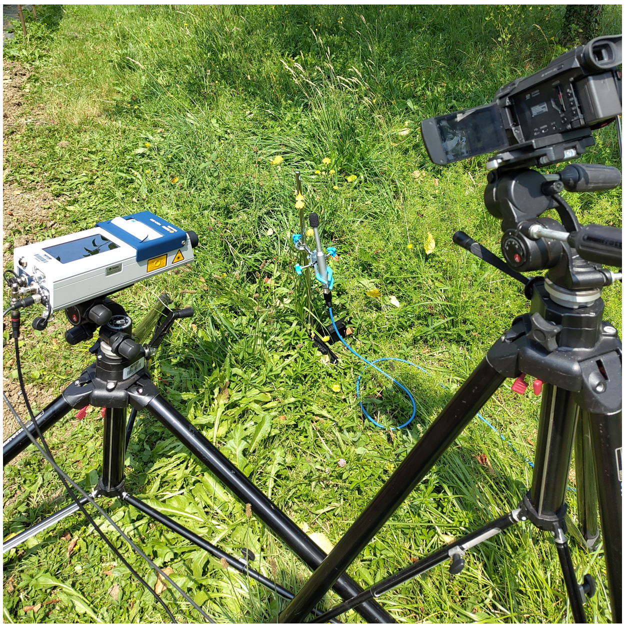


Figure 1: Data acquisition setup for recording vibration signals, audio, and visual recordings from flowers.

Labeled intervals were cut into clips of one second with 0.1-second overlap (positive instances), whereas unlabeled portions were similarly divided and treated as negative instances. To balance the dataset, the negative instances were randomly downsampled. Some negative instances contained environmental noise, such as speech, machinery, or wind, and wind noise was occasionally present in positive instances. Examples of vibration signals from honey bee foraging and from wind are shown in Figures 2 and 3, respectively. The final balanced subset consisted of 7334 positive and 8664 negative instances. The positive data distribution by insects is given in Table 1.

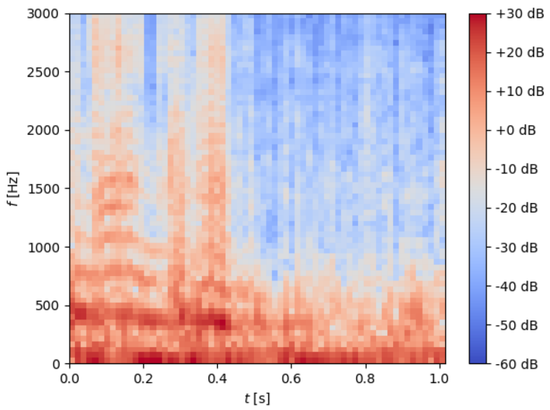


Figure 2: Sample spectrogram of honey bee foraging (positive)

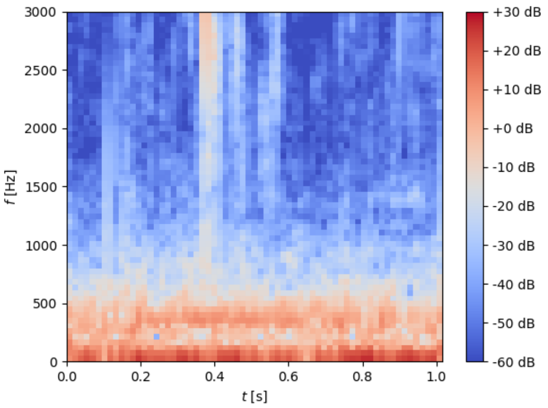


Figure 3: Sample spectrogram of light wind blowing (negative)

Table 1: Number of labels and the corresponding number of instances by insect.

Insect	Number of labels	Instances
fly	76	4146
honey bee	253	1688
wild bee	98	1307
hoverfly	82	155
bumble bee	14	24
wasp	3	9
moth	1	5
Total	527	7334

3 Methodology

The objective of this study was to assess whether stem vibrations can be used to detect the presence of pollinators on flowers. From a machine learning perspective, the problem was framed as a binary classification task: distinguishing between the presence and absence of insects in physical contact with the flower. The methodology consisted of initial recoding of waveforms and labeling, preprocessing the data, selecting the appropriate neural network architecture, and training and evaluating the model.

3.1 Data Preprocessing

First, the instances that were shorter than one second (in cases where the expert-labeled interval was shorter than one second) were padded. After that, all instances were then converted into Mel spectrograms of size 64x64 using fast Fourier transform with frequency range 0–3 kHz.

3.2 Model Architecture

For the model, a network of residual blocks in combination with convolution was used. It is a smaller version of some ResNet (e.g. ResNet 18) models. Residual blocks offer efficient reuse of features across the layers. As shown in Figure 4, the input spectrogram goes through a 3x3 convolution, followed by three residual blocks, before final pooling. The residual block, shown in Figure 5, consists of two 3x3 convolutions to identify features and residual path only uses stride to match the shape before addition.

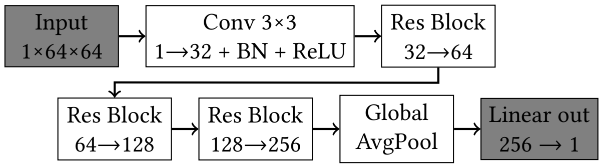


Figure 4: Model Architecture

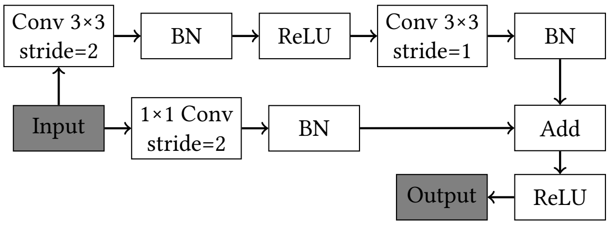


Figure 5: Residual Block (Res Block) in Figure 4

To prevent overfitting and to enable extended training, dropout of 0.5 was used, which improved performance more than data augmentation (and was also computationally more efficient).

3.3 Model Training Settings

The model was trained by using the binary cross-entropy loss. Optimization was performed with Adam optimizer with learning rate 10^{-4} and weight decay 10^{-5}. A batch size of 16 and an epoch number of 30 were used.

4 Evaluation Metrics

We used standard performance evaluation metrics: accuracy, precision, recall and F1-score, which were computed from the number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{1}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{F1-score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

In confusion matrices, we used relative numbers samples for colors instead of absolute (which are only listed), because there was much more negatives than positives in detection test. Relative shares are based on true labels (e.g. fraction of FN among all negatively labeled).

4.1 Experiments

The model was evaluated in three experimental settings, all using 5-fold cross-validation. Instances originating from the same recording were always assigned to the same fold to better reflect real-world variability. Training and testing were repeated five times, each with a different fold held out for testing and the remaining folds used for training. Reported results are averages across the five folds.

4.1.1 Balanced labeled subset. In the first experiment, called "Subset", only the manually labeled subset of the dataset was used. This consisted of the 7334 positive and 8664 negative instances as described above. These were treated as balanced binary classification data and evaluated directly.

4.1.2 Full dataset with raw labeling. In the second experiment, called "Full data (raw)", the entire dataset was included by segmenting recordings into 1.0 s windows with a step size of 0.1 s. Expert annotations were then used to assign labels to these windows, yielding a much larger evaluation set. However, such raw labeling frequently introduced short, isolated positive or negative events that were likely erroneous. When the model predicted such isolated events, performance metrics were underestimated, as the evaluation framework treated them as genuine labels. This motivated the introduction of a heuristic smoothing procedure.

4.1.3 Full dataset with heuristic labeling. The third experiment, called "Full data (heuristics)", used the same sliding-window segmentation as raw labeling experiment, but applied a heuristic smoothing procedure to adjust labels. The aim was to reduce the influence of short, likely erroneous events while preserving longer, fragmented signals as single detections. Two rules were applied:

- If the model predicted at least 10 consecutive positive windows (equivalent to 1.0 s), the entire interval was relabeled as positive.
- If at least 82% of 50 consecutive windows (equivalent to 5.0 s) were predicted as positive, the entire interval was relabeled as positive.

These empirically determined thresholds suppressed short false positives while ensuring that extended pollinator events with intermittent weak signals were still detected as continuous segments. Finally, because the sliding window (1.0 s) exceeded the step size (0.1 s), prediction timestamps were shifted backward by 0.5 s to align the window centers with the expert annotations.

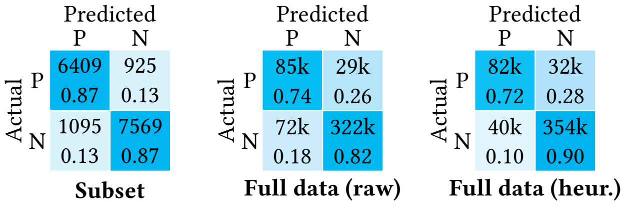
5 Results and Discussion

The results of all three experiments are shown in Table 2 along with the confusion matrices in Figure 6.

Table 2: Results of all experiments. The numbers represent the average ± standard deviation across five folds in the cross-validation.

	Subset	Full data (raw)	Full data (heur.)
Accuracy	0.87 ± 0.03	0.80 ± 0.02	0.86 ± 0.05
Precision	0.85 ± 0.09	0.54 ± 0.11	0.68 ± 0.15
Recall	0.87 ± 0.04	0.75 ± 0.11	0.73 ± 0.13
F1-score	0.86 ± 0.06	0.62 ± 0.07	0.69 ± 0.11

The results show that there was a significant reduction in performance when we switched from using the balanced subset to recordings from the full dataset. There are several possible sources of error: labels are annotated on waveform and samples are extracted in the way that the whole non-padded (therefore non-silent) part is either positive either negative, furthermore, prediction for a specific time t is generated based on the window, beginning at t and ending at $t + 1$ s, which might lead to inaccuracies at edges of labels although we shifted the time to match it as good as possible. There are also no other insects or activities in samples, which occur in full recordings and are



Subset

Actual \ Predicted	P	N
P	6409 0.87	925 0.13
N	1095 0.13	7569 0.87

Full data (raw)

Actual \ Predicted	P	N
P	85k 0.74	29k 0.26
N	72k 0.18	322k 0.82

Full data (heur.)

Actual \ Predicted	P	N
P	82k 0.72	32k 0.28
N	40k 0.10	354k 0.90

Figure 6: Confusion matrices of all 3 experiments, described in section 4.1

sometimes falsely positive. It is important to note that the "Full data" is not a balanced set (while "Subset" is) and is meant as a test for a real-world scenario, where conditions and frequency of pollinators with them vary on short time scales (hours), which makes loss balancing (which would reduce the gap between recall and precision) in practice very difficult. For this reason, we did not use it and we left the thresholds the same as in the "Subset" experiment, so the results serve as a valid estimation of the performance in reality. Figure 7 shows how heuristics helped the model by smoothing out the short erroneous predictions, resulting in improved performance. To improve model performance even further, additional heuristic filters may be added.

6 Conclusion

We presented initial results on the feasibility of detecting pollinator presence on flowers from stem vibration recordings using machine learning methods. We evaluated models under three experimental settings: a balanced labeled subset, the full dataset with raw expert annotations, and the full dataset with heuristic label smoothing. The results demonstrate that pollinator activity can be reliably inferred from vibration signals, with heuristic post-processing substantially reducing the impact of isolated erroneous predictions and improving the robustness of detection.

Future work will focus on extending the models beyond binary detection towards classification of pollinator species and potentially of behavioral activities. From an applied perspective, the long-term goal is to develop lightweight vibration detectors that can be mounted directly on plants to automatically register pollinator visits. Deploying a small number of such sensors in a field or meadow would enable scalable estimation of pollinator abundance and activity, providing a valuable tool for biodiversity monitoring and conservation studies.

Acknowledgements

The authors acknowledge the funding from the Slovenian Research and Innovation Agency, Grants PR-10495, P1-0255, J7-50040, Z1-50018, the Basic core funding P2-0209, and the support received from the Erasmus+ Traineeship programme.

References

[1] European Commission. Joint Research Centre. 2021. *Proposal for an EU pollinator monitoring scheme.* Publications Office, LU. DOI: 10.2760/881843.

[2] David L. Wagner. 2020. Insect declines in the anthropocene. *Annual Review of Entomology*, 65, 1, (Jan. 2020), 457–480. DOI: 10.1146/annurev-ento-011019-025151.

[3] Roel van Klink, Diana E. Bowler, Konstantin B. Gongalsky, Ann B. Swengel, Alessandro Gentile, and Jonathan M. Chase. 2020. Meta-analysis reveals declines in terrestrial but increases in freshwater insect abundances. *Science*, 368, 6489, (Apr. 2020), 417–420. DOI: 10.1126/science.aax9931.

[4] J. C. Biesmeijer et al. 2006. Parallel declines in pollinators and insect-pollinated plants in britain and the netherlands. *Science*, 313, 5785, (July 2006), 351–354. DOI: 10.1126/science.1127863.

[5] Luísa Gigante Carvalheiro et al. 2013. Species richness declines and biotic homogenisation have slowed down for <scp>nw</scp>-european pollinators and plants. *Ecology Letters*, 16, 7, (May 2013), 870–878. Yvonne Buckley, editor. DOI: 10.1111/ele.12121.

[6] David Susič, Johanna A. Robinson, Danilo Bevk, and Anton Gradišek. 2025. Acoustic monitoring of solitary bee activity at nesting boxes. *Ecological Solutions and Evidence*, 6, 3, e70080. e70080 ESO-24-09-164.R1. eprint: https://besjournals.onlinelibrary.wiley.com/doi/pdf/10.1002/2688-8319.70080. DOI: https://doi.org/10.1002/2688-8319.70080.

[7] Rok Šturm, Juan José López Díez, Jernej Polajnar, Jérôme Sueur, and Meta Virant-Doberlet. 2022. Is it time for ecotremology? *Frontiers in Ecology and Evolution*, 10, (Mar. 2022). DOI: 10.3389/fevo.2022.828503.

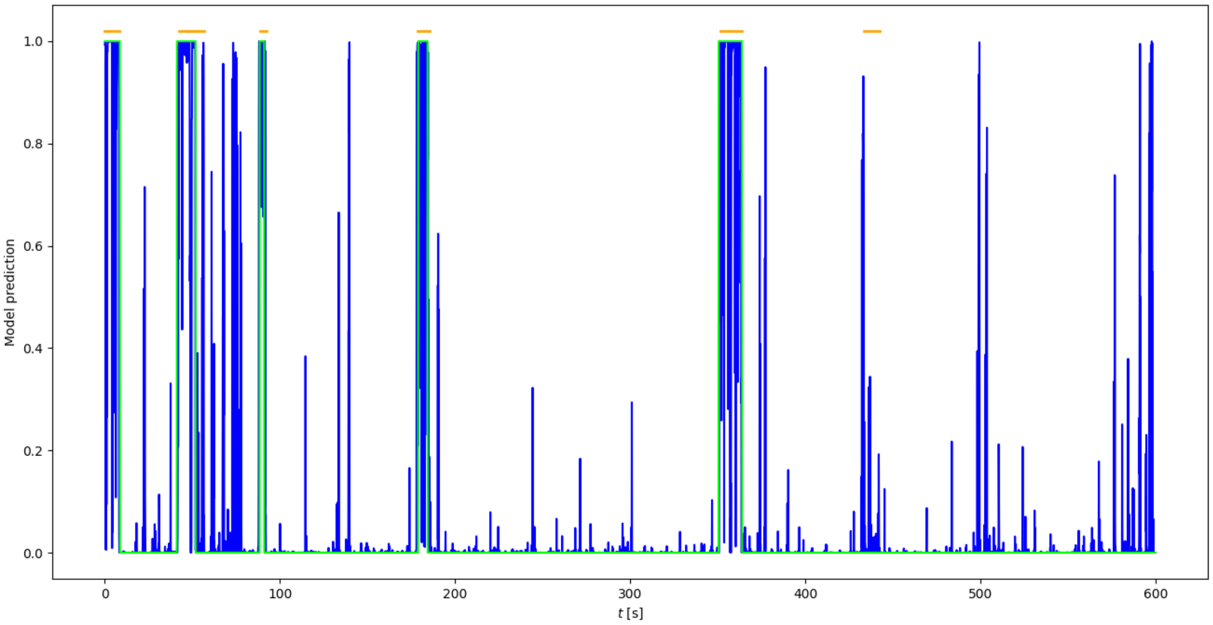


Figure 7: Output example: (blue) model prediction, (green) heuristic filter, (yellow) expert labels.

Thermal Camera-Based Cognitive Load Estimation: A Non-Invasive Approach

Zoja Anžur
zoja.anzur@ijs.si
Jožef Stefan Institute
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia

Gašper Slapničar
gasper.slapnicar@ijs.si
Jožef Stefan Institute
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia

Mitja Luštrek
mitja.lustrek@ijs.si
Jožef Stefan Institute
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia

Abstract

Cognitive load (CL) monitoring is a growing area of interest across various domains. Most traditional methods rely on either subjective assessments or intrusive sensors, limiting their practical applicability. In this study, we present a non-invasive approach for estimating CL using thermal imaging. Thermal videos were collected from 18 participants performing a battery of tasks designed to induce varying levels of CL. Using a low-cost thermal camera, we extracted features from facial regions of interest and trained several machine learning models, including Random Forest, Extreme Gradient Boosting, Stochastic Gradient Descent (SGD), k-Nearest Neighbors, and Light Gradient Boosting Machine, on a binary classification task distinguishing between rest and high CL conditions. The models were evaluated using Leave-One-Subject-Out cross-validation. Our results show that all models outperform the baseline majority classifier, with SGD achieving the highest accuracy (0.64 ± 0.16), despite variability across individuals. These findings support the feasibility of thermal imaging as an unobtrusive tool for CL estimation in real-world applications.

Keywords

cognitive load estimation, thermal imaging, physiological computing, machine learning for affective computing, non-invasive user monitoring

1 Introduction

Monitoring cognitive load (CL) unobtrusively and accurately has become an increasingly important goal across various domains. Traditional methods such as the NASA-TLX questionnaire [7] for assessing cognitive states often rely on intrusive sensors or subjective self-reporting, limiting their practicality in real-world applications. In recent years, the use of machine learning techniques combined with physiological signals has opened new possibilities for non-invasive and continuous monitoring [2].

The primary objective of our study was to predict CL using data obtained with a thermal camera. Our aim was to develop a method for unobtrusive measurement of physiological signals that achieves high accuracy. Compared to other physiological measurement tools, thermal cameras are relatively low-cost and quick to deploy, which makes them a practical choice for real-world cognitive monitoring applications.

2 Related Work

Early approaches of contact-free thermal monitoring of psychophysiological states based on infrared thermal imaging focused primarily on emotional and affective research [8]. Physiological background was heavily explored, specifically how autonomic nervous system activity yields descriptive thermal signatures related to affect in facial regions. Such work laid the critical groundwork for later expansion towards CL estimation.

One of the fundamental studies towards thermal-camera-based CL estimation was published by Abdelrahman et al. in 2017. They introduced an unobtrusive method that uses a commercial thermal camera to monitor temperature changes on the forehead and nose, which were chosen as regions of interest based on physiological background established earlier. It demonstrated that the difference between forehead and nose temperature correlates robustly with task difficulty, showing effectiveness in Stroop test and reading complexity experiments. Notably, the system achieved near-real-time detection with an average latency of 0.7 seconds, making it suitable for responsive, real-time cognition-aware applications [1].

While such monitoring traditionally required relatively expensive hardware [6], recent work showed potential of more affordable low-cost thermal cameras for monitoring of psychological states. Black et al. [4] compared state-of-the-art vision transformers (ViT) against traditional convolutional neural networks (CNNs) on data recorded with low-resolution thermal cameras. They found superior performance of ViT when classifying emotions, achieving 0.96 F1 score for 5 emotions (anger, happiness, neutral, sadness, surprise), confirming feasibility of low-cost hardware.

Lastly, some work explores subtle connections between different inner states that are difficult to discriminate, such as stress and CL. Bonyad et al. [5] showed correlation of the two states in airplane pilots, highlighting that elevated cognitive workload induced stress, manifesting in significant cooling across the nose, forehead, and cheeks, with the nasal region exhibiting the most rapid and pronounced temperature decline. These thermal changes were synchronized with increases in heart rate and subjective workload ratings. Overall thermal monitoring is becoming more accessible and an established CL estimation alternative to other modalities (e.g., wearables, RGB cameras, etc.), especially in challenging conditions (e.g., darkness).

3 Data

3.1 Data Collection

For the purpose of our experiment, we gathered data from 18 participants using various sensors. In this work, we will focus only on relevant data obtained by an affordable FLIR Lepton 3.5

Information Society 2025, Ljubljana, Slovenia

https://doi.org/10.70314/is.2025.skui.3714

camera, with resolution of 160x120 running at 8.7 frames per second.

Our participants underwent a battery of tests for inducing CL. Data collection was carried out in a controlled laboratory environment to ensure consistency across all participants. After filling out some initial questionnaires regarding individual's tiredness and focus levels, the calibration of various sensors used in the study was performed. The experiment itself was structured into three sequential blocks, each designed to induce CL through two different tasks offered at two difficulty levels. The first block featured standardized CL tasks – specifically, the N-back and Stroop tasks, which are widely used in cognitive research to engage working memory and executive attention [10, 12].

The second block introduced more ecologically valid memory tasks. The memory recall task involved displaying a list of words on a screen, after which participants had 30 seconds to recall and verbally report as many as possible. In the visual memory task, participants observed an image and were later asked to recall specific details.

The third and final block focused on ecological visual attention tasks. These included a visual discrepancy detection task and a line tracking task. In the discrepancy detection task, participants compared two images and identified visual differences. In the line tracking task, participants followed numbered lines from one side of the screen to the other and identified them.

Between these cognitive tasks, participants engaged in relaxation activities such as resting, passively viewing images, or listening to music, which served as baseline conditions and helped to balance their CL throughout the experiment. After each task and each relaxation period, participants completed the NASA Task Load Index (NASA-TLX) [7] and the Instantaneous Self-Assessment (ISA) [9] questionnaires to provide subjective evaluations of their cognitive and affective states.

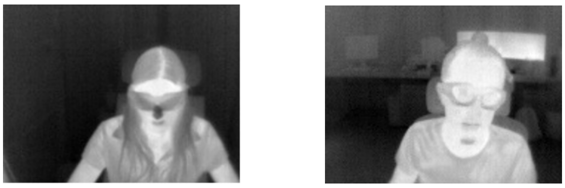
The session concluded with the removal of all sensors, a debriefing session, and participant compensation. The entire procedure lasted approximately 60 minutes per participant, with around 40 minutes spent for active data collection and the rest used for setup, instructions, and debriefing.

3.2 Data Preprocessing

The raw data used in our analysis is illustrated in Figure 1. The first step in our preprocessing pipeline was windowing. Specifically, we divided each thermal video into consecutive 3-second windows with a 25% overlap. From each window, only the middle frame was selected for further analysis. This approach was based on the assumption that facial temperature changes driven by physiological responses such as blood flow occur gradually over several seconds rather than instantaneously. As such, a single representative frame from each interval was considered sufficient to capture meaningful thermal variation in 2.25-second steps.

The second step in preparing the data for subsequent machine learning experiments involved the extraction of features from thermal camera recordings. Prior research in this domain frequently utilizes average temperatures from distinct facial regions as input features, demonstrating that these regions can exhibit significant temperature differences associated with various affective states experienced by participants [3]. Motivated by these findings, we adopted a similar methodology to that proposed by Aristizabal-Tique et al. [3], and based our feature set on the average temperatures of four predefined regions of interest (ROIs): nose, forehead, left eye, and right eye.

The first step in obtaining the average temperatures for the selected ROIs involved applying a facial keypoint detector to extract the coordinates corresponding to each region in the thermal images. This process was carried out for the middle frame of every window of the thermal videos by passing it through a pretrained keypoint detection model [11]. The model, based on the widely adopted YOLOv5 architecture, was specifically trained on thermal images to enhance its performance for this modality. Following keypoint detection, we transitioned from working with raw thermal images to working with numerical temperature features, specifically the average temperatures computed for each region of interest. A more detailed explanation of this feature extraction process is provided in Section 4.1.



(a) Subject A.

(b) Subject B.

Figure 1: Examples of raw thermal images.

At this stage, our dataset – where each row corresponded to a single video frame – contained a substantial number of missing values. These missing values were primarily due to limitations in keypoint detection, which stemmed from several factors. First, participants wore smart glasses during the experiment, which often obstructed the eye region and impaired the accuracy of the keypoint detector. Second, natural head movements, such as turning to the left or right, occasionally caused parts of the face to be occluded, preventing the detector from accurately identifying key facial landmarks. Given the impact of these issues on data quality, we chose to remove all rows containing missing values from further analysis. We excluded 31% of the data in this step. Use of smart glasses was not problematic only for keypoint detection, but also for feature calculation. The eye regions were partially obstructed by the glasses, thus preventing the thermal camera from capturing accurate temperature measurements in this area. Since we were unable to control for this effect, it is possible that it also posed an issue in classification.

Next, we performed label transformations to prepare the data for subsequent analysis. Initially, the dataset included multiple labels, each corresponding to one of the tasks described in Section 3.1. However, approximately 50% of the instances were labeled as "questionnaire", reflecting the periods during which participants completed self-report instruments such as NASA-TLX and ISA. These instances posed a challenge: filling out a questionnaire is neither a clear resting state nor a cognitively demanding task, making it difficult to accurately determine the level of CL involved. Since our primary interest lay in distinguishing between load and rest conditions, we opted to exclude all rows labeled as "questionnaire" from further analysis. In addition, we grouped the remaining labels into three broader categories: rest, low CL (corresponding to the easy versions of the tasks), and high CL (corresponding to the difficult versions).

Following some initial experiments, we chose to retain only the most "extreme" instances in terms of CL. Specifically, we excluded all data labeled as low CL, as this class exhibited substantial overlap with both the rest and high load conditions. In particular, some tasks intended to induce low CL turned out to be unexpectedly difficult, effectively eliciting high CL, while others were so easy that it is questionable whether they imposed any cognitive demand at all.

To further emphasize the most distinct cognitive states, we also filtered the remaining data within each label interval. For intervals of instances labeled as rest, we retained only the final two-thirds of each interval, based on the assumption that participants would be most physiologically relaxed toward the end of each interval labeled rest. Immediately after completing a cognitively demanding task, the body may require some time to "cool down", during which residual physiological activity – such as elevated facial temperature – could still be present. By focusing on the latter portion of the interval, we aimed to capture a more accurate representation of the true resting state. Similarly, for instances labeled as high CL, we also retained only the final two-thirds of each interval, based on the assumption that CL tends to accumulate toward the end of a demanding task. This selection strategy was intended to maximize the contrast between rest and high load conditions by focusing on the time points most representative of those states.

4 Methodology

4.1 Calculating Features

As previously mentioned, we extracted features directly from the raw thermal images. Using the pretrained keypoint detector [11], we obtained coordinates for five facial keypoints, using which we then defined ROIs corresponding to specific facial areas for each 3-second window. ROIs were shaped as rectangles, positioned based on keypoint coordinates. Their size and placement were dynamically defined according to the distance between the eyes, reducing issues such as capturing inconsistent facial areas due to variations in distance from the camera or head movements. This approach was considered appropriate, because the study was conducted in a controlled laboratory environment with minimal variation in posture and setup. Additionally, a visual inspection of the extracted ROIs confirmed that they were well aligned.

Next, we computed the average pixel temperature for each ROI, as each pixel in a thermal image directly reflects a temperature value. This process yielded four primary features – one for each of the predefined ROIs (nose, forehead, left eye, and right eye). To capture relative temperature differences between these regions, we then computed the pairwise differences between all four average temperatures. This resulted in an additional six features, representing the thermal contrasts between different facial areas. Finally, to capture potential temporal trends in temperature changes, we introduced two additional temporal features. Specifically, for each 3-second window, we computed the temperature difference between the first and last frame for two key regions of interest: the nose and the forehead. These temporal features aimed to reflect short-term thermal dynamics that may be indicative of CL fluctuations. In total, this process resulted in 12 features per instance: 4 average ROI temperatures, 6 pairwise temperature differences, and 2 temporal difference features.

Finally, we applied personalized normalization to account for individual differences in baseline physiological responses. While there is considerable variability across participants, the variations within each individual are more informative for detecting changes in CL. To address this, we standardized all feature values using z-score normalization per participant, transforming each instance based on that individual's mean and standard deviation. This normalization helped reduce inter-subject variability while preserving intra-subject dynamics, enabling a more robust learning of patterns related to CL. Following this step, we proceeded with the machine learning experiments using the described set of features.

Table 1: Class distribution

Label	Count
Rest	1626
High Load	1548

4.2 Experiments

After completing the data preparation steps outlined in Sections 3.2 and 4.1, we proceeded with the machine learning experiments. At this stage, the dataset consisted of two balanced classes: rest and high CL, as shown in Table 1. The models were trained on a total of 3174 instances, derived from 18 participants.

In our experiments, we employed a diverse set of machine learning models, including Random Forest (RF), Extreme Gradient Boosting (XGB), Stochastic Gradient Descent (SGD), k-Nearest Neighbors (KNN), and Light Gradient Boosting Machine (GBM). As a baseline, we included a majority classifier, which always predicted the most frequent class in the training data of each fold. Each model was trained and evaluated using its optimized hyperparameters, which were determined through a grid search strategy applied on training data on each Leave-One-Subject-Out (LOSO) iteration aimed at maximizing classification accuracy.

To ensure the robustness and generalizability of the results, we adopted a LOSO cross-validation approach, in which each participant served as a test subject exactly once while the remaining participants were used for training. This evaluation strategy is well-suited for personalized and physiological data, where inter-subject variability is high. To ensure a comprehensive evaluation of model performance, we did not rely solely on a single metric. Instead, we incorporated a range of evaluation metrics, including accuracy and F1-score. This multi-metric approach allowed us to better capture different aspects of model performance. The results of these experiments are presented in the subsequent section.

5 Results

As mentioned in the previous sections, we trained and evaluated a variety of models, and evaluated them using the LOSO. Summary of the results can be seen in Table 2, where both accuracy and F1-score are reported as averages across all subject folds, providing an overall measure of model performance and generalization performance.

The results indicate that the best-performing algorithm was SGD, achieving an accuracy of 0.64 ± 0.16, which represents a 0.13 improvement over the baseline majority classifier accuracy of 0.51 ± 0.00. In addition to its accuracy, SGD also achieved a high F1-score, suggesting that the model performs well in predicting both classes in a balanced manner. However, SGD also has the highest variance (± 0.16), which indicates less stability across subjects. Overall, all evaluated models outperformed the majority class baseline. Moreover, the accuracy scores across all tested models were relatively similar, indicating consistent performance regardless of the specific algorithm used. Performance of GBM,

Table 2: Accuracy and F1-score of trained models compared to the majority class classifier

Classifier	Model Accuracy	Model F1	Majority Class Accuracy	Majority Class F1
RF	0.62 ± 0.13	0.62 ± 0.13	0.51 ± 0.00	0.34 ± 0.04
XGB	0.62 ± 0.14	0.62 ± 0.14	0.51 ± 0.00	0.34 ± 0.04
SGD	**0.64 ± 0.16**	**0.63 ± 0.16**	**0.51 ± 0.00**	**0.34 ± 0.04**
KNN	0.60 ± 0.10	0.60 ± 0.10	0.51 ± 0.00	0.34 ± 0.04
GBM	0.63 ± 0.10	0.60 ± 0.11	0.51 ± 0.00	0.34 ± 0.04

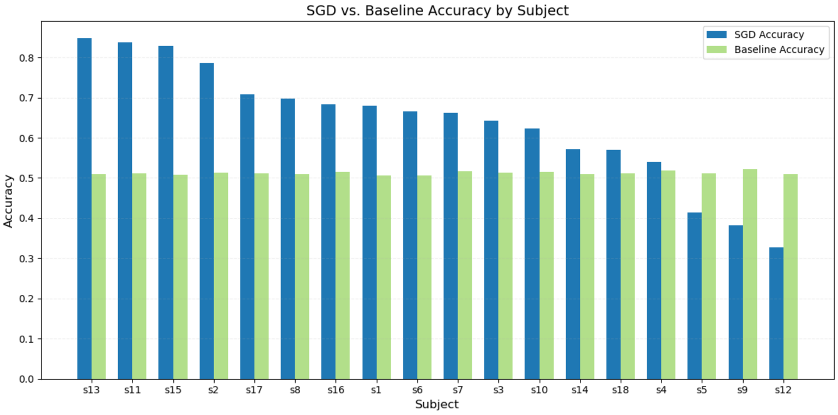


Figure 2: SGD vs. baseline majority classifier by subject.

RF and XGB was very similar, although somewhat behind the performance of the SGD.

Looking at per-subject results in more detail in Figure 2, we see that for most subjects, the SGD classifier outperformed the majority baseline classifier. SGD achieved its best performance on subjects 13, 11, and 15, with accuracies exceeding 0.80. There is also considerable variation across individuals, which aligns with the high variance reported in Table 2. This variability may indicate the presence of subject-specific patterns, label noise, or data that is inherently more challenging to learn.

6 Conclusion

This study demonstrates the potential of low-cost consumer thermal imaging as a viable, non-invasive method for estimating CL. By leveraging features extracted from key facial regions and applying various machine learning algorithms, we achieved promising results in distinguishing between rest and high load cognitive states. Among the tested models, SGD achieved the best average performance, though with notable inter-subject variability. These findings highlight both the strengths and current limitations of thermal-based CL estimation. While the results support the feasibility of using affordable thermal cameras in real-world applications, future work should explore strategies such as more sophisticated personalization to enhance generalization across individuals, deep learning, etc. This line of research points toward usefulness of cognitive monitoring in practical settings such as education, workplace safety, and adaptive user interfaces.

Acknowledgements

We sincerely thank our colleagues from Department of Intelligent Systems (Jožef Stefan Institute) for their assistance in data collection and preprocessing.

References

[1] Yomna Abdelrahman, Eduardo Velloso, Tilman Dingler, Albrecht Schmidt, and Frank Vetere. 2017. Cognitive heat: exploring the usage of thermal imaging to unobtrusively estimate cognitive load. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1, 3, 1–20.

[2] Muneeb Imtiaz Ahmad, Ingo Keller, David A Robb, and Katrin S Lohan. 2023. A framework to estimate cognitive load using physiological data. *Personal and Ubiquitous Computing*, 27, 6, 2027–2041. DOI: 10.1007/s00779-020-01455-7.

[3] Victor H. Aristizabal-Tique, Marcela Henao-Pérez, Diana Carolina López-Medina, Renato Zambrano-Cruz, and Gloria Díaz-Londoño. 2023. Facial thermal and blood perfusion patterns of human emotions: proof-of-concept. *Journal of Thermal Biology*, 112, 103464. DOI: https://doi.org/10.1016/j.jtherbio.2023.103464.

[4] James Thomas Black and Muhammad Zeeshan Shakir. 2025. Ai enabled facial emotion recognition using low-cost thermal cameras. *Computing&AI Connect*, 2, 1, 1–10.

[5] Amin Bonyad, Hamdi Ben Abdessalem, and Claude Frasson. 2025. Heat of the moment: exploring the influence of stress and workload on facial temperature dynamics. In *International Conference on Intelligent Tutoring Systems*. Springer, 181–193.

[6] Federica Gioia, Maria Antonietta Pascali, Alberto Greco, Sara Colantonio, and Enzo Pasquale Scilingo. 2021. Discriminating stress from cognitive load using contactless thermal imaging devices. In *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 608–611.

[7] Sandra G Hart and Lowell E Staveland. 1988. Development of nasa-tlx (task load index): results of empirical and theoretical research. In *Advances in psychology*. Vol. 52. Elsevier, 139–183.

[8] Stephanos Ioannou, Vittorio Gallese, and Arcangelo Merla. 2014. Thermal infrared imaging in psychophysiology: potentialities and limits. *Psychophysiology*, 51, 10, 951–963.

[9] CS Jordan and SD Brennen. 1992. Instantaneous self-assessment of workload technique (isa). *Defence Research Agency, Portsmouth*.

[10] Michael Kane, Andrew Conway, Timothy Miura, and Gregory Colflesh. 2007. Working memory, attention control, and the n-back task: a question of construct validity. *Journal of experimental psychology. Learning, memory, and cognition*, 33, (May 2007), 615–22. DOI: 10.1037/0278-7393.33.3.615.

[11] Askat Kuzdeuov, Dana Aubakirova, Darina Koishigarina, and Huseyin Atakan Varol. 2022. Tfw: annotated thermal faces in the wild dataset. *IEEE Transactions on Information Forensics and Security*, 17, 2084–2094. DOI: 10.1109/TIFS.2022.3177949.

[12] Michael P Milham, Kirk I Erickson, Marie T Banich, Arthur F Kramer, Andrew Webb, Tracey Wszalek, and Neal J Cohen. 2002. Attentional control in the aging brain: insights from an fmri study of the stroop task. *Brain and cognition*, 49, 3, 277–296.

A Critical Perspective on MNAR Data: Imputation, Generation, and the Path Toward a Unified Framework

Fatemeh Azad
fatemeh.azad@fri.uni-lj.si
University of Ljubljana
Ljubljana, Slovenia

Matjaž Kukar
matjaz.kukar@fri.uni-lj.si
University of Ljubljana
Ljubljana, Slovenia

Abstract

Missing Not at Random (MNAR) data remains one of the most difficult challenges in statistical analysis and machine learning. Despite the widespread availability of advanced imputation methods, most research continues to focus on Missing Completely at Random (MCAR) and partially on Missing at Random (MAR) scenarios. This paper provides a critical overview of existing approaches for MNAR imputation, methods for simulating MNAR data, and the limitations of current evaluation practices. We highlight the lack of standardized benchmarks, unrealistic missingness rates, and insufficient coverage of MNAR conditions in empirical studies. Finally, we propose a suitable framework for comprehensive testing of design principles, enabling robust and reproducible evaluation of imputation methods across mechanisms and missingness rates.

Keywords

Missing data, MNAR, data imputation, missingness mechanisms, data generation, machine learning, evaluation framework.

1 Introduction

Missing data is a pervasive challenge across various domains, from clinical diagnostics and bioinformatics to finance, sensor networks, and social sciences. Missing, damaged, or unrecorded data entries can negatively affect the accuracy of statistical analysis and machine learning models. They reduce predictive power, introduce bias, and often create incompatibilities with algorithms requiring complete inputs [8]. The impact is especially important in critical areas like healthcare decision support, where unreliable data or incorrect interpretation can lead to harmful conclusions.[14, 2].

A primary difficulty in handling missing data is understanding the underlying *missingness mechanism*. According to the taxonomy of Little and Rubin [10], We have three types of missingness: *Missing Completely at Random* (MCAR), *Missing at Random* (MAR), and *Missing Not at Random* (MNAR).

To formally describe the MCAR, MAR, and MNAR mechanisms, we first define the following notation, as per [9, 19]:

X**:** the complete data matrix, which consists of two parts, with X_{obs} being the observed and X_{mis} the missing part of the data.

R**:** an indicator matrix of the same dimensions as X, where $R_{ij} = 1$ if the value X_{ij} is missing, and $R_{ij} = 0$ if it is observed.

ψ**:** a parameter or set of parameters that govern the missingness process.

- Data is *MCAR* if the probability of a value being missing is completely independent of both the observed and the unobserved data. The missingness is unrelated to the data itself — it is a purely random (Eq. 1) as the missingness pattern (R) depends neither on the observed data (X_{obs}) nor on the missing data (X_{mis}).

$$P(R|X_{obs}, X_{mis}, \psi) = P(R|\psi) \tag{1}$$

- Data is *MAR* if the probability of a value being missing depends only on the observed data, not on the missing data itself (Eq. 2). This means that the missingness could be predicted from available (non-missing) data. The probability of the missingness pattern (R) is conditionally independent of the actual missing values (X_{mis}) once the observed values (X_{obs}) are taken into account.

$$P(R|X_{obs}, X_{mis}, \psi) = P(R|X_{obs}, \psi) \tag{2}$$

- Data is *MNAR* if the probability of a value being missing depends on some unobserved (missing) value itself, even after accounting for all the observed data (Eq. 3). In this case (X_{mis}) can also include latent features, unobserved for all instances. This is the most complex scenario, as the missingness pattern itself is informative. The probability of the missingness pattern (R) is therefore dependent on the missing values (X_{mis}) in a way that cannot be explained by the observed values (X_{obs}).

$$P(R|X_{obs}, X_{mis}, \psi) \tag{3}$$

While MCAR and MAR have been extensively studied, MNAR remains the most difficult and least explored scenario, precisely because the missingness itself carries information about the data. For example, high-income individuals may systematically withhold reporting their wealth, or patients with severe conditions may drop out of longitudinal studies. In both cases, the very act of non-response encodes meaningful but hidden signals.

The prevalent imputation (replacing missing values) research has focused on MCAR and MAR settings, where assumptions about independence or conditional dependence simplify methodological development and evaluation [14, 23, 13]. In contrast, MNAR scenarios pose a dual challenge: not only is the missing information inherently dependent on unobserved values, but there are also very few benchmark datasets that explicitly model or annotate MNAR mechanisms. Consequently, evaluation standards remain incomplete. Reported missingness rates often underestimate or ignore MNAR effects, and even sophisticated models, such as generative adversarial networks [7, 24], graph neural approaches [25], or transformer-based imputers [3], rarely demonstrate systematic robustness in MNAR conditions. Recent works [11, 4] have shown the potential of ensemble or meta-imputation strategies, which combine diverse imputers

Information Society 2025, Ljubljana, Slovenia

https://doi.org/10.70314/is.2025.skui.0957

into robust pipelines. However, these frameworks are also mostly validated under MCAR or MAR assumptions.

In this paper, we take a critical perspective on the current state of missing data research, specifically focusing on MNAR. We argue that three gaps must be addressed: (i) the lack of effective imputation techniques designed specifically for MNAR, as current methods are limited in scope and seldom used in practice; (ii) the deficiency of datasets and generators that can faithfully represent MNAR patterns; and (iii) the insufficiency of reported missingness rates. To bridge these gaps, we outline the vision and design principles of a comprehensive *framework* for MNAR research that integrates data generation, imputation, and evaluation under standardized conditions. Such a framework would enable more robust comparisons of existing methods and guide the development of novel techniques tailored to the inherent challenges of MNAR.

The remainder of this paper is organized as follows. Section 2 reviews existing imputation approaches and discusses their applicability to MNAR. Section 3 examines methodologies for simulating and generating MNAR data, highlighting their limitations. Section 4 critiques how missingness is reported and motivates the need for standardized benchmarks. Finally, Section 5 presents our vision for a unified MNAR research framework and outlines open challenges for the community.

2 Imputation Methods for MNAR Data

A wide range of imputation techniques has been proposed in the literature, from simple statistical to advanced deep generative models. While these methods have demonstrated effectiveness under Missing Completely at Random (MCAR) or Missing at Random (MAR) assumptions, their suitability for Missing Not at Random (MNAR) scenarios remains highly questionable. This section reviews the main categories of imputation techniques and highlights their limitations when faced with MNAR data.

While it is often stated that there are almost no methods tailored for MNAR, several strands of work do exist … However, these remain underutilized and rarely integrated into mainstream imputation pipelines.

2.1 Statistical Imputation Methods

Statistical techniques such as mean, median, mode, or regression-based imputations are simple and computationally efficient but they mostly rely on strong assumptions about the independence or conditional dependence of missingness [8, 27]. These assumptions rarely hold under MNAR, where the missingness mechanism is informative itself. For example, imputing systematically underreported values (e.g., income, clinical severity) with central-tendency statistics introduces bias and distorts the true distribution. Maximum likelihood and Bayesian approaches attempt to capture uncertainty, but they typically assume that the missingness process can be ignored or is fully modeled by observed data [10], which is not the case for MNAR.

2.2 Machine Learning-Based Approaches

Machine learning methods, such as k-nearest neighbors (KNN) [14], matrix factorization [20], decision trees [21], and support vector machines (SVMs) [6], utilize feature dependencies to address missing data entries. While more flexible than statistical methods, they fail when the missingness depends on unobserved or latent variables. For instance, if severely ill patients systematically omit follow-up surveys, no observed features can explain this absence, and machine learning based imputers cannot recover the missing structure without explicitly modeling the mechanism.

2.3 Deep Learning Approaches

Deep generative models have significantly advanced imputation research. Variational Autoencoders (VAEs) [2] and Generative Adversarial Networks (GANs) [23, 7, 24] are capable of learning complex distributions and have shown robustness to high missingness rates. However, their performance in the context of MNAR conditions is not assured. While some frameworks, such as MisGAN, explicitly attempt to learn the missingness mask distribution alongside the data [7], they often rely on approximations that do not generalize across domains. Similarly, diffusion-based models [22, 26] and graph-based imputers [25] extend coverage to structured data but rarely test systematically against MNAR conditions. Transformers, such as ReMasker [3], provide context-aware imputations, but again, their evaluations are mostly limited to MCAR and MAR scenarios.

2.4 Ensemble Approaches

Recent efforts highlight the potential of combining multiple imputers in ensemble or meta-learning frameworks [11, 4]. Such methods leverage complementary strengths of diverse imputers and often achieve more stable performance across heterogeneous datasets. However, existing ensemble frameworks have been validated primarily under MCAR assumptions, and their ability to handle MNAR remains largely unexplored. Recent work has also explored meta-imputation strategies, such as the Meta-Imputation Balanced (MIB) framework, which combines multiple base imputers in a supervised setting [1].

To synthesize the discussion above, Table 1 summarizes the main categories of imputation approaches, their representative methods, applicability to missingness mechanisms, and key references.

3 Generation of MNAR Data

A persistent challenge in missing data research is the lack of reliable and reproducible benchmarks for handling MNAR scenarios. While MCAR and MAR can be easily simulated by random masking or conditioning on observed features, MNAR requires masking rules that depend on unobserved or latent variables, which makes the generation process more challenging. Consequently, most experimental studies rely on oversimplified masking strategies that do not capture the complexity of real-world MNAR mechanisms [18, 5].

3.1 The Role of Data Amputation

Deliberately injecting missing values into fully complete datasets, referred to as *data amputation*, plays a crucial role in evaluating imputation techniques. However, until recently, implementations of amputation were highly heterogeneous and often insufficiently documented, preventing fair comparisons across studies [18]. This problem is particularly acute for MNAR, where even slight differences in implementation can lead to very different conclusions.

Table 1: Comparison of Imputation Approaches from Literature

Approach	Representative Methods	Missingness Types Addressed	Representative References
Traditional Statistical	Mean, Median, Mode, Regression-based, Maximum Likelihood, Bayesian Approaches	MCAR only (rarely MAR)	Schafer & Graham [27], Little & Rubin [10], Lin & Tsai [8]
Machine Learning	KNN, Matrix Factorization, Decision Trees, SVM	MCAR, partially MAR	Murti et al. [14], Lee et al. [20], Song & Lu [21], Feng et al. [6]
Deep Learning	VAEs, GANs, Diffusion Models, Graph-based Models, Transformers	MCAR, MAR (limited MNAR)	Collier et al. [2], Yoon et al. [23, 24], Li et al. [7], Du et al. [3], Tashiro et al. [22], Zheng & Charoenphakdee [26], You et al. [25]
Meta-Learning / Ensembles	Meta Learning, Meta-Regressio, MIB Frameworkn	MCAR, partially MAR; potential for MNAR	Liu et al. [11], Ellington et al. [4], Azad et al. [1]

3.2 Artificial MNAR Generation Strategies

The most common way to simulate MNAR is by masking values as a function of their own magnitude or distribution. For instance, removing a feature's highest or lowest values mimics non-disclosure of extreme outcomes (e.g., very high glucose levels) [18]. Stochastic variants extend this idea by assigning missingness probabilities proportional to the unobserved value itself, enabling flexible control over the intensity of missingness [16]. While intuitive, such strategies remain oversimplified, often restricted to univariate rules that fail to capture the multi-dimensional dependencies of real domains [5].

Recent work has proposed standardized libraries for data amputation to address reproducibility concerns. The `mdatagen` package provides a broad set of implementations for MCAR, MAR, and MNAR, supporting univariate and multivariate scenarios [12]. In particular, it incorporates advanced MNAR mechanisms such as Missingness Based on Own Values (MBOV), Missingness Based on Own and Unobserved Values (MBOUV), and Missingness Based on Intra-Relations (MBIR) [15]. These implementations move beyond ad hoc thresholding by systematically encoding missingness processes and offering reproducible pipelines. In addition, `mdatagen` includes visualization and evaluation modules, allowing researchers to inspect missingness patterns and assess their impact on imputation performance.

Together, these synthetic and standardized approaches form the current toolkit for MNAR data generation. However, despite their usefulness, they remain abstractions of real-world processes and should ideally be complemented by domain-informed simulations.

3.3 Domain-Inspired Simulation

Beyond standardized libraries, domain knowledge remains critical for realistic MNAR generation. In healthcare, dropout is often linked to disease severity, side effects, or socioeconomic constraints. In socioeconomic surveys, non-response may be strongly correlated with privacy-sensitive attributes such as income or debt. Encoding these mechanisms requires integrating causal assumptions with probabilistic masking rules [17]. However, such domain-specific approaches are difficult to generalize, limiting their utility as benchmarks.

4 Toward a Unified Framework for MNAR Research

Two key insights emerge from the previous sections: (i) current imputation methods are not explicitly designed for MNAR, and (ii) the lack of realistic MNAR generators inhibits effective evaluation. To address these gaps, we anticipate a unified framework integrating generation, imputation, and evaluation of MNAR data under standardized and reproducible conditions.

4.1 Design Principles

A comprehensive MNAR framework should have the following principles:

- **Synthetic realism:** Data generators should simulate MNAR scenarios that mirror real-world domains (e.g., systematic dropout in healthcare, self-censoring in socio-economic data), either by extending existing functionality (e.g., mdatagen [12]) or by incorporating custom plug-in modules. To balance interpretability with scalability, both threshold-based rules and learned mechanisms should be supported.
- **Comprehensive evaluation:** Benchmarks must test across all three missingness mechanisms (MCAR, MAR, MNAR) and a full spectrum of missingness rates.
- **Cross-domain applicability:** The framework should support diverse data types (tabular, sequential, multimodal) and allow integration of domain knowledge for context-specific MNAR simulation.

4.2 Proposed Framework Components

We propose that a unified MNAR framework should consist of three interdependent modules:

1. **MNAR Data Generators:** Domain-informed and probabilistic tools for simulating missingness patterns that depend on latent or unobserved values, using existing libraries ([12] or incorporating custom plug-in functions.
2. **Imputation Engines:** A modular interface with plug-in adapters for existing methods that support statistical, machine learning, deep learning, and ensemble methods [14, 23, 1]. By isolating imputers within a common framework, researchers can test their robustness under controlled MNAR scenarios.
3. **Evaluation Suite:** Standardized protocols that combine direct metrics (e.g., Mean Absolute Error (MAE), Root Mean Squared Error (RMSE)) with indirect metrics (downstream predictive performance, such as accurracy, RMSE/MAE, or domain relevant metrics such as interpretability, reliability, fairness, . . .) [1].

4.3 Benefits and Impact

Developing such a framework would enable several advances:

- *Reproducibility:* Common benchmarks and generators ensure that different imputation methods can be fairly compared.
- *Realism:* Domain-specific MNAR mechanisms bring evaluations closer to real-world conditions, reducing the gap between research and practice.
- *Innovation:* By exposing the weaknesses of existing methods under MNAR, the framework incentivizes the development of mechanism-aware imputers.
- *Generalization:* Unified treatment of MCAR, MAR, and MNAR encourages methods that adapt to unknown or mixed missingness mechanisms without prior assumptions.

5 Conclusion

Missing data remains one of the most persistent challenges in machine learning and statistical analysis. While decades of research have produced numerous imputation techniques, ranging from simple statistical estimators to deep generative models, most methods have been designed and evaluated under the more tractable MCAR and MAR mechanisms. In contrast, the most realistic and challenging setting, MNAR, remains critically underexplored.

Our review highlights three major gaps in the current state of the field. First, existing imputation methods rarely model the dependence of missingness on unobserved values, making them unsuitable for MNAR scenarios. Second, generating realistic MNAR data is crucial because most benchmarks use ad hoc or overly simplistic masking strategies, which fail to capture the complexity of real-world missingness. Third, evaluation standards remain incomplete, with reported missingness rates often conflating MCAR/MAR assumptions and failing MNAR realities. Together, these shortcomings hinder fair comparisons and limit methodological innovation.

To address these challenges, we propose the vision and design principles of a unified MNAR framework that integrates three components: (i) data generators that are aware of mechanisms and can create realistic MNAR patterns, (ii) modular imputation engines that enable thorough testing of various methods, and (iii) extensive evaluation suites that include direct metrics and indirect metrics. Such a framework would provide reproducibility, realism, and a strong foundation for developing next-generation imputation techniques.

Future research should move toward principled, mechanism-aware imputers and adopt standardized benchmarks for MNAR generation and evaluation. To advance MNAR research, we need more powerful algorithms and standardized tools and protocols that enhance rigor and comparability in the field.

Acknowledgements

The research and development presented in this paper were funded by the Research Agency of the Republic of Slovenia (ARIS) through the ARIS Young Researcher Programme (research core funding No. P2-209). While preparing this work, the authors used Grammarly to check the correctness of grammar and improve the fluency of the writing, aiming to enhance the clarity and impact of the publication. The authors reviewed and edited the content produced with this tool/service and accept full responsibility for the final published content.

References

[1] Fatemeh Azad, Zoran Bosnić, and Matjaž Kukar. 2025. Meta-imputation balanced (mib): an ensemble approach for handling missing data in biomedical machine learning. In *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine (BIBE)*. submitted.

[2] Mark Collier, Bayan Mustafa, and Mihaela van der Schaar. 2020. VAEs in the presence of missing data. *arXiv:2006.05301*.

[3] Meng Du, Gábor Melis, and Zhaozhi Wang. 2023. Remasker: imputing tabular data with masked autoencoding. In *The Eleventh International Conference on Learning Representations*.

[4] E. Ellington, Guillaume Bastille-Rousseau, Cayla Austin, Kristen Landolt, Bruce Pond, Erin Rees, Nicholas Robar, and Dennis Murray. 2014. Using multiple imputation to estimate missing data in meta-regression. *Methods in Ecology and Evolution*, 6, (Dec. 2014). DOI: 10.1111/2041-210X.12322.

[5] Tlamelo Emmanuel, Thabiso Maupong, Dimane Mpoeleng, Thabo Semong, Banyatsang Mphago, and Oteng Tabona. 2021. A survey on missing data in machine learning. (May 2021). DOI: 10.21203/rs.3.rs-535520/v1.

[6] Hao Feng, Lihui Chen, and Ke Wang. 2005. A svm regression based approach to filling in missing values. In *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*. Springer, 581–587.

[7] Shun-Chuan Li, Bingsheng Jiang, and Benjamin M Marlin. 2019. Misgan: learning from incomplete data with generative adversarial networks. In *International Conference on Learning Representations*.

[8] Wei-Chao Lin and Chih-Fong Tsai. 2020. Missing value imputation: a review and analysis of the literature (2006–2017). *Artificial Intelligence Review*, 53, 1487–1509.

[9] Roderick J. A. Little and Donald B. Rubin. 1986. *Statistical Analysis with Missing Data*. John Wiley & Sons. ISBN: 978-0471802545.

[10] Roderick JA Little and Donald B Rubin. 2019. *Statistical analysis with missing data*. Vol. 793. John Wiley & Sons.

[11] Qian Liu and Manfred Hauswirth. 2020. A provenance meta learning framework for missing data handling methods selection. In *2020 11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*. DOI: 10.1109/UEMCON51285.2020.9298089.

[12] Arthur Mangussi, Miriam Santos, Filipe Loyola Lopes, Ricardo Cardoso Pereira, Ana Lorena, and Pedro Henriques Abreu. 2025. Mdatagen: a python library for the artificial generation of missing data. *Neurocomputing*, 625, (Apr. 2025), 129478. DOI: 10.1016/j.neucom.2025.129478.

[13] Pierre-Alexandre Mattei and Jes Frellsen. 2019. Miwae: deep generative modelling and imputation of incomplete data sets. In *International conference on machine learning*. PMLR, 4413–4423.

[14] Dinar M P Murti, I N A Ramatryana, and A P Wibawa. 2019. K-nearest neighbor (k-nn) based missing data imputation. In *2019 5th International Conference on Science in Information Technology (ICSITech)*. IEEE, 83–88.

[15] 2023. *Siamese autoencoder-based approach for missing data imputation.* (June 2023), 33–46. ISBN: 978-3-031-35994-1. DOI: 10.1007/978-3-031-35995-8_3.

[16] 2023. *Automatic delta-adjustment method applied to missing not at random imputation.* (June 2023), 481–493. ISBN: 978-3-031-35994-1. DOI: 10.1007/978-3-031-35995-8_34.

[17] Ricardo Cardoso Pereira, Joana Cristo Santos, José Amorim, Pedro Rodrigues, and Pedro Henriques Abreu. 2020. Missing image data imputation using variational autoencoders with weighted loss. In (Apr. 2020).

[18] Miriam Seoane Santos, Ricardo Cardoso Pereira, Adriana Fonseca Costa, Jastin Pompeu Soares, João Santos, and Pedro Henriques Abreu. 2019. Generating synthetic missing data: a review by missing mechanism. *IEEE Access*, 7, 11651–11667. DOI: 10.1109/ACCESS.2019.2891360.

[19] Joseph L. Schafer and John W. Graham. 2002. Missing data: our view of the state of the art. *Psychological methods*, 7 2, 147–77. https://api.semanticscholar.org/CorpusID:7745507.

[20] Nandana Sengupta, Madeleine Udell, Nathan Srebro, and James Evans. 2022. Sparse data reconstruction, missing value and multiple imputation through matrix factorization. *Sociological Methodology*.

[21] Ying-Ying Song and Ying Lu. 2015. Decision tree methods: applications for classification and prediction. *Shanghai archives of psychiatry*, 27, 2, 130.

[22] Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. 2021. Csdi: conditional score-based diffusion models for probabilistic time series imputation. In *Advances in Neural Information Processing Systems*. Vol. 34, 24804–24816.

[23] Jinsung Yoon, James Jordon, and Mihaela van der Schaar. 2018. GAIN: missing data imputation using generative adversarial nets. In *International conference on machine learning*. PMLR, 5689–5698.

[24] Sanghoon Yoon and Sanghoon Sull. 2020. Gamin: generative adversarial multiple imputation network for highly missing data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8456–8464.

[25] Jiaxuan You, Xiaobai Ma, Daisy Ding, Mykel Kochenderfer, and Jure Leskovec. 2020. Handling missing data with graph representation learning. In *Advances in Neural Information Processing Systems*. Vol. 33, 18357–18368.

[26] Shuhan Zheng and Nontawat Charoenphakdee. 2022. Diffusion models for missing value imputation in tabular data. *arXiv preprint arXiv:2210.17128*.

[27] Yuyang Zhou, Sarjyt Aryal, and Mohamed Reda Bouadjenek. 2024. Review for handling missing data with special missing mechanism. *arXiv preprint arXiv:2404.04905*.

Utilizing Large Language Models for Supporting Multi-Criteria Decision Modelling Method DEX

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

Uroš Rajkovič
Faculty of Organizational Sciences
University of Maribor
Kranj, Slovenia
uros.rajkovic@um.si

Vladislav Rajkovič
Faculty of Organizational Sciences
University of Maribor
Kranj, Slovenia
vladislav.rajkovic@gmail.com

Abstract

We experimentally assessed the capabilities of two mainstream artificial intelligence chatbots, ChatGPT and DeepSeek, to support the multi-criteria decision-making process. Specifically, we focused on using the method DEX (Decision EXpert) and investigated their performance in all stages of DEX model development and utilization. The results indicate that these tools may substantially contribute in the difficult stages of collecting and structuring decision criteria, and collecting data about decision alternatives. However, at the current stage of development, the support for the whole multi-criteria decision-making process is still lacking, mainly due to occasionally inconsistent and erroneous execution of methodological steps.

Keywords

Multi-criteria decision-making, decision analysis, large language models, method DEX (Decision EXpert), structuring decision criteria

1 Introduction

Multi-criteria decision-making (MCDM) [1] is an established approach to support decision-making in situations where it is necessary to consider multiple interrelated, and possibly conflicting criteria, and select the best solution based on the available alternatives and the preferences of the decision-maker. Traditionally, such models are developed in collaboration with decision makers and domain experts, who define the criteria, acquire decision makers' preferences and formulate the corresponding evaluation rules. The model-development process is demanding, as it includes structuring the problem, formulating all the necessary model components (such as decision preferences or rules) for evaluating decision alternatives, and analyzing the results.

With the development and success of generative artificial intelligence, especially large language models (LLMs) [2], the question arises as to how these models can support or perhaps partially automate decision-making processes. To this end, we explored the capabilities of recent mainstream LLM-based chatbots, specifically ChatGPT and DeepSeek, for supporting the MCDM process. We specifically focused on using the method DEX (Decision EXpert) [3], with which we have extensive experience, spanning multiple decades [4], in the roles of decision makers, decision analysts, and teachers. DEX is a full-aggregation [5] multi-criteria decision modelling method, which proceeds by making an explicit decision model. DEX uses qualitative (symbolic) variables to represent decision criteria, and decision rules to represent decision makers' preferences. Variables (attributes) are structured hierarchically, representing the decomposition of the decision problem into smaller, easier to handle subproblems. Traditionally, DEX models are developed using software such as DEXiWin [6], which helps the user to interactively construct a DEX model and use it to evaluate and analyze decision alternatives.

The reported research is of exploratory nature. We ran ChatGPT and DeepSeek multiple times over the last six months, either individually, as a group or in classrooms with students. Typically, we first formulated some hypothetical decision problem and then guided the chatbot through the main stages of the MCDM process:

A. Model development stages:
   1. Acquiring criteria
   2. Definition of attributes (variables representing criteria)
   3. Structuring attributes
   4. Preference modeling (formulating decision rules)

B. Model utilization stages:
   5. Definition of decision alternatives
   6. Evaluation of alternatives
   7. Explaining the results of evaluation
   8. Analysis of alternatives

In doing this, we observed the responses generated by the LLMs and assessed them from the viewpoint of skilled decision analysts. The main goal was not to solve specific real-life decision problems, but to identify LLMs' strengths and weaknesses that may substantially affect the MCDM process.

Despite focusing on DEX, many of our findings are also applicable to other hierarchical full-aggregation MCDM methods [1, 5], such as AHP, MAUT/MAVT, and MACBETH, which follow the same methodological stages, with slight differences in the representation of model components.

In the following sections, we review the above-mentioned MCDM stages and describe our experience with each of them. Specifically, we illustrate the process with answers generated by ChatGPT-o3 and DeepSeek-V3. We considered a hypothetical personal decision problem of buying an electric-powered vehicle

Information Society 2025, 6–10 October 2025, Ljubljana, Slovenia

http://doi.org/10.70314/is.2025.skui.6589

(EV). The chatbots were run in parallel on June 6th, 2025, using similar prompts. Our assessments and comments are somewhat broader, based on some other use-cases, not presented here.

2 Acquiring Criteria

A MCDM model includes multiple criteria that capture essential aspects of decision alternatives. These criteria are used to evaluate and compare the alternatives in a structured way. Defining criteria usually requires a good knowledge of the decision problem and the decision maker's goals. This step is usually one of the most difficult and may require consulting domain experts and/or relevant literature.

We asked the chatbots: *I am considering buying a new EV vehicle for up to 35000 EUR in Slovenia. Suggest criteria for evaluating such cars.*

Both ChatGPT and DeepSeek came out with an extensive and structured list of criteria. Table 1 shows the criteria suggested by DeepSeek. ChatGPT's suggestions were similar, though they employed slightly different high-level categories, which also incorporated measurement units associated with each criterion.

Table 1: EV cars: Criteria suggested by DeepSeek. Items 4–8 shortened due to space limitations.

1. Range & Battery Efficiency

- **Real-world range** (WLTP or EPA estimates) – Aim for **at least 300 km** in mixed conditions.
- **Battery size** (kWh) – Larger batteries offer more range but may increase cost.
- **Energy consumption** (kWh/100 km) – Lower is better (e.g., 14–18 kWh/100 km).

2. Charging Speed & Compatibility

- **DC fast-charging capability** (e.g., 50 kW, 100 kW, or higher).
- **AC charging speed** (e.g., 7.4 kW, 11 kW, or 22 kW for home/work charging).
- **Charging port type** (CCS Combo is standard in Europe).

3. Performance & Driving Experience

- **Motor power** (kW/HP) – Affects acceleration and highway performance.
- **Drivetrain** (FWD, RWD, AWD) – AWD is better for Slovenia's varied weather.
- **Top speed & acceleration** (0–100 km/h time).

4. Practicality & Usability:

- **Cargo space, Seating capacity, Ground clearance**

5. Price & Incentives

- **Purchase price, Slovenian EV incentives, Insurance & maintenance costs**

6. Technology & Features

- **Infotainment system, Driver assistance, Heat pump availability**

7. Availability & After-Sales Support

- **Local dealership & service network, Warranty**

8. Resale Value & Future-Proofing

- **Brand reputation**, **Software updates**

Let us immediately say that we consider this the most important single contribution of LLMs to MCDM modeling. We are not aware of any previous method that would allow identifying and structuring decision criteria in such a depth and detail in literally just a minute. Of course, for "serious" applications getting such a list does not take the burden off the user, who is still responsible for verifying the suggestions and checking the criteria for relevance and correctness. Nevertheless, this is a valuable starting point that can save days or even weeks of work. This stage does not depend on the MCDM method used, so other methods may benefit from using LLMs equally well.

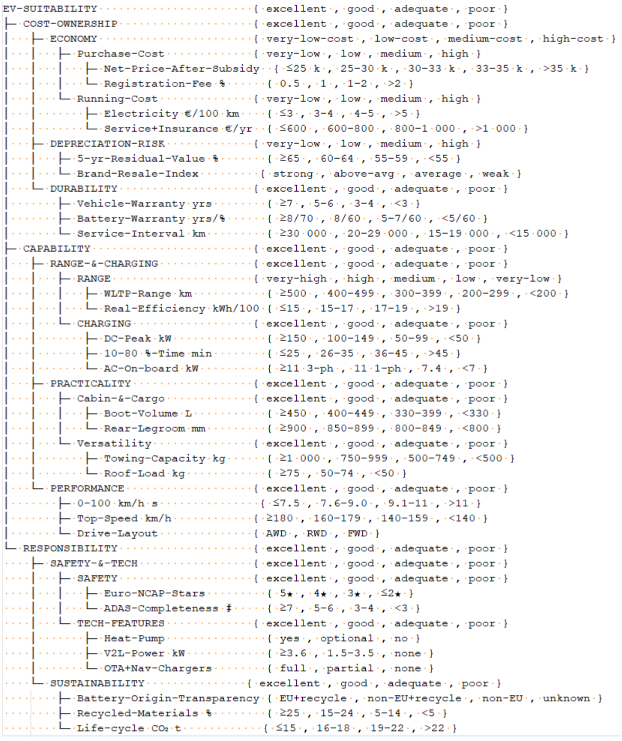
3 Definition of Attributes

In this stage, the task is to define variables, called attributes, that represent criteria in a MCDM model. As most MCDM methods use numeric attributes, this stage is specific to DEX, which uses qualitative attributes. Therefore, this and the following stages require LLMs to "understand" the method used. While DEX is less widely known than methods such as AHP or MAUT, it is nonetheless used and valued in various applications. Anyway, we were somewhat surprised to find out that all consulted LLMs were familiar with DEX and reasonably capable of following its main methodological steps. In some steps, however, we had to specify additional requirements to obtain proper DEX model components.

Generally, defining qualitative value scales of attributes was not too difficult for LLMs. Asking *Suggest preferentially ordered value scales* typically gives good suggestions for value scales, for example (DeepSeek):

Purchase Price: High (>€34k) → Medium (€30k–34k) → Low (<€30k)
Government Incentives: None → Moderate (€1k–3k) → High (>€3k)
Insurance & Maintenance: Expensive → Moderate → Cheap

Interestingly, in our first attempts at using LLMs, they were not fully aware of DEX specifics, such as that value scales should have a small number of values, preferentially ordered from "bad" to "good" [3]. After correcting LLMs a few times, they "remembered" and now suggest properly formulated value scales.



```
EV-SUITABILITY                         { excellent , good , adequate , poor }
├─ COST-OWNERSHIP                      { excellent , good , adequate , poor }
│   ├─ ECONOMY                         { very-low-cost , low-cost , medium-cost , high-cost }
│   │   ├─ Purchase-Cost               { very-low , low , medium , high }
│   │   │   ├─ Net-Price-After-Subsidy  { ≤25 k , 25-30 k , 30-33 k , 33-35 k , >35 k }
│   │   │   └─ Registration-Fee %       { 0.5 , 1 , 1-2 , >2 }
│   │   └─ Running-Cost                { very-low , low , medium , high }
│   │       ├─ Electricity €/100 km     { ≤3 , 3-4 , 4-5 , >5 }
│   │       └─ Service+Insurance €/yr   { ≤600 , 600-800 , 800-1 000 , >1 000 }
│   ├─ DEPRECIATION-RISK               { very-low , low , medium , high }
│   │   ├─ 5-yr-Residual-Value %        { ≥65 , 60-64 , 55-59 , <55 }
│   │   └─ Brand-Resale-Index          { strong , above-avg , average , weak }
│   └─ DURABILITY                      { excellent , good , adequate , poor }
│       ├─ Vehicle-Warranty yrs         { ≥7 , 5-6 , 3-4 , <3 }
│       ├─ Battery-Warranty yrs/%       { ≥8/70 , 8/60 , 5-7/60 , <5/60 }
│       └─ Service-Interval km          { ≥30 000 , 20-29 000 , 15-19 000 , <15 000 }
├─ CAPABILITY                          { excellent , good , adequate , poor }
│   ├─ RANGE-&-CHARGING                { excellent , good , adequate , poor }
│   │   ├─ RANGE                       { very-high , high , medium , low , very-low }
│   │   │   ├─ WLTP-Range km            { ≥500 , 400-499 , 300-399 , 200-299 , <200 }
│   │   │   └─ Real-Efficiency kWh/100  { ≤15 , 15-17 , 17-19 , >19 }
│   │   └─ CHARGING                    { excellent , good , adequate , poor }
│   │       ├─ DC-Peak kW               { ≥150 , 100-149 , 50-99 , <50 }
│   │       ├─ 10-80 %-Time min         { ≤25 , 26-35 , 36-45 , >45 }
│   │       └─ AC-On-board kW           { ≥11 3-ph , 11 1-ph , 7.4 , <7 }
│   ├─ PRACTICALITY                    { excellent , good , adequate , poor }
│   │   ├─ Cabin-&-Cargo               { excellent , good , adequate , poor }
│   │   │   ├─ Boot-Volume L            { ≥450 , 400-449 , 330-399 , <330 }
│   │   │   └─ Rear-Legroom mm          { ≥900 , 850-899 , 800-849 , <800 }
│   │   └─ Versatility                 { excellent , good , adequate , poor }
│   │       ├─ Towing-Capacity kg       { ≥1 000 , 750-999 , 500-749 , <500 }
│   │       └─ Roof-Load kg             { ≥75 , 50-74 , <50 }
│   └─ PERFORMANCE                     { excellent , good , adequate , poor }
│       ├─ 0-100 km/h s                { ≤7.5 , 7.6-9.0 , 9.1-11 , >11 }
│       ├─ Top-Speed km/h              { ≥180 , 160-179 , 140-159 , <140 }
│       └─ Drive-Layout                { AWD , RWD , FWD }
└─ RESPONSIBILITY                      { excellent , good , adequate , poor }
    ├─ SAFETY-&-TECH                   { excellent , good , adequate , poor }
    │   ├─ SAFETY                      { excellent , good , adequate , poor }
    │   │   ├─ Euro-NCAP-Stars          { 5★ , 4★ , 3★ , ≤2★ }
    │   │   └─ ADAS-Completeness #      { ≥7 , 5-6 , 3-4 , <3 }
    │   └─ TECH-FEATURES               { excellent , good , adequate , poor }
    │       ├─ Heat-Pump                { yes , optional , no }
    │       ├─ V2L-Power kW             { ≥3.6 , 1.5-3.5 , none }
    │       └─ OTA+Nav-Chargers         { full , partial , none }
    └─ SUSTAINABILITY                 { excellent , good , adequate , poor }
        ├─ Battery-Origin-Transparency { EU+recycle , non-EU+recycle , non-EU , unknown }
        ├─ Recycled-Materials %         { ≥25 , 15-24 , 5-14 , <5 }
        └─ Life-cycle CO₂ t            { ≤15 , 16-18 , 19-22 , >22 }
```

Figure 1: DEX model structure generated by ChatGPT.

4 Structuring Attributes

After acquiring the criteria and defining attributes, the next step is to structure attributes into a hierarchy (most often an ordinary tree). Following the previous stages, which already resulted in a well-organized criteria structure and proper definition of attributes, this stage looks quite easy for LLMs. Figure 1 shows a full DEX model structure as suggested by ChatGPT without any further instructions. In comparison, the DeepSeek's suggestion (not shown here) was somewhat inferior. Also, it was generated only after we had instructed it that DEX requires "narrow" trees with only two to three descendants for each aggregate attribute. Anyway, we consider both structures appropriate and comparable to the achievements of the best university-level students.

5 Decision Preferences and Decision Rules

In DEX, decision maker's preferences are represented by decision rules, organized in decision tables. For each aggregate attribute, the user is asked to investigate all combinations of lower-level attribute values and assess the corresponding outcomes. Depending on the number of aggregate attributes, this might be a laborious task, but can be usually carried out using software such DEXiWin with not too much hassle. LLMs are also capable of suggesting perfectly valid decision tables, as illustrated in Table 2, which suggest the values of *Purchase-Cost* depending on *Net-Price-After-Subsidy* and *Registration-Fee*.

Table 2: Decision table for Purchase-Cost (ChatGPT).

Net-Price-After-Subsidy ▼ / Registration-Fee ►	**very-low (0 .5 %)**	**low (1 %)**	**medium (1–2 %)**	**high (> 2 %)**
very-low (≤ 25 k €)	very-low	low	low	medium
low (25–30 k €)	low	low	medium	medium
medium (30–33 k €)	medium	medium	medium	high
high (33–35 k €)	high	high	high	high

From the DEX perspective, it is important to remark that Table 2 is complete (addressing all possible combinations of input values) and preferentially consistent (increasing input values result in increasing outputs). Initially, these requirements were not obvious to LLMs, and we had to request them explicitly. After further use, LLMs now generate appropriate rules by themselves.

However, it is important to emphasize an essential issue. Decision making is considered a subjective process, and decisions should reflect individual decision-makers' preferences. Using LLMs, we only get preferences, generalized from many documents. In order to impose our own preferences, we should (1) carefully check LLMs' suggestions and (2) request modifications, if necessary. According to our experience, this is possible by formulating prompts such as "*increase the importance of* some attribute" or "*alternatives exceeding* some value *of* some attribute *are unacceptable*". Nevertheless, this requires focus, a good understanding of the goals, and perpetual verification of the results.

6 Identification of Decision Alternatives

The goal of this stage is to identify decision alternatives and describe them in terms of values, assigned to input attributes of the model. This is another stage in which LLMs excel due to their ability to act like agents: autonomously collecting relevant data from external sources. In this case, the LLMs constructed lists of actual EV cars, consulting sources such as avto.net, citroen.si, reuters.com, and ev-database.org:

MG 4 Comfort 64 kWh – €31 490 → €24 290 after Eko-sklad

Citroën ë-C4 54 kWh – €31 900 → €24 700

Opel Corsa Electric 54 kWh – from €28 490 → €21 290

BYD Dolphin Surf Boost 43 kWh – €22 990 → €15 790*

Fiat 500e 42 kWh – promo €22 290 (price already after bonus)

It is important to understand that alternatives should be defined after developing the model. This allows the LLM to search for and collect the data needed to evaluate the alternatives.

7 Evaluation of Alternatives

In this stage, the five cars have been evaluated by the LLMs. All were evaluated as GOOD, except Fiat 500e, which was evaluated as ADEQUATE. Table 3 illustrates the output generated by ChatGPT, showing the evaluation results of the two best-ranked cars. In addition to the overall evaluation (EV-SUITABILITY), the output includes evaluations achieved at the three main branches of the model, together with a brief notification of reasons for such evaluations. This is an appropriate summary of evaluation results.

Table 3: Evaluation of the two best-ranked EV cars (ChatGPT).

Model	**MG 4 Comfort 64**		**Citroën ë-C4**	
Cost-Ownership	**good**	(low purchase-cost; running-cost low)	**good**	(very-low purchase-cost & running-cost)
Capability	**good**	(435 km WLTP, 140 kW DC, 363 L boot)	**adequate–good**	(416 km WLTP, 100 kW DC, 380 L boot)
Responsibility	**good**	(★ 5 Euro-NCAP, heat-pump optional, V2L 3.6 kW)	**good**	(★ 4 Euro-NCAP, heat-pump std., forthcoming V2L)
EV-SUITABILITY	**GOOD**		**GOOD**	
Rank	1		2	

8 Explaining the Results of Evaluation

In MCDM, it is essential to understand the evaluation method and explain the reasons underlying evaluation results. This is how ChatGPT explains the evaluation method:

1. Leaf values were taken from the most recent Slovene price-lists or Europe-wide tech sheets (all ≤ €35 000 trims).
2. Running-costs use your 30 000 km/year mileage and the Q1-2025 average household tariff of €0.166 kWh-1.
3. Each quantitative item was mapped to the ordered value-scales we set earlier; the three rule-tables ("purchase-cost", "running-cost", "economy") were then fired, and all higher nodes were combined with simple equal weights (0.4 Cost-Ownership | 0.4 Capability | 0.2 Responsibility).

4. Scores were coded excellent = 4, good = 3, adequate = 2, poor = 1; the root class was chosen by rounding the weighted mean to the nearest integer.

At first, everything seems to make sense. However, careful reading reveals the disturbing fact that different decision rules from those agreed upon in the previous stages were used to evaluate alternatives. Unfortunately, this often happens with LLMs, which tend to "forget" about the previous MCDM stages. It is not uncommon that attributes, their value scales and decision rules change from prompt to prompt. This severely undermines the trust in using LLMs and makes the whole process uneasy: rather than focusing on solving the decision problem, the user is forced to meticulously check each and every step. Also, it is not uncommon to discover logical errors or even basic computational errors (often referred to as "hallucinations" [7]). In one of our sessions with ChatGPT, it displayed the evaluation formula

$$(0.2\times3)+(0.25\times4)+(0.15\times4)+(0.2\times3)+(0.15\times2)+(0.05\times2)=3.15$$

which looked convincing, but gave a hard-to-notice, but wrong result; the correct result is 3.2.

9 Analysis of Alternatives

The last stage of the MCDM process is the analysis of alternatives, which is aimed at exploring the decision space using methods such as what-if and sensitivity analysis. Without providing experimental evidence due to space restrictions, we can say that, in principle, LLMs are capable of performing such analyses, giving appropriate answers and explanations to questions such as:

- *Carry out sensitivity analysis for Citroën ë-C3 and MG4 depending on buying price and operating costs.*
- *What would have to change for Fiat 500e 42 to become a good EV vehicle?*

In most cases, results are correct and informative, particularly in cases when an explicit explanation is requested by the user. However, the issues of using inappropriate model components and making logical and computational errors were detected in this stage as well.

10 Discussion

LLMs are developing rapidly and becoming increasingly capable. They may evolve under the hood, so that even the same version can behave differently depending on recent updates or user-specific factors. This makes them challenging for conducting a rigorous scientific research. They come without user manuals, requiring their users to explore their capabilities on their own. This study is an experimental attempt to understanding the capabilities of the current (2025) mainstream LLMs for supporting the MCDM process, with special emphasis on the DEX method. On this basis, we could not formulate firm conclusions, but were still able to make observations and formulate recommendations that might help MCDM practitioners.

The single most important contribution of LLMs to MCDM is their ability to formulate a well-structured list of relevant criteria in the first stage (section 2). Nothing nearly as good was available so far for that difficult stage, where LLMs can now substantially boost the process and save a lot of effort and time. The second important contribution is the capability of LLMs to act as agents and collect data about alternatives (section 6) from various external resources.

Considering individual MCDM stages, LLMs performance is quite impressive. They are capable of evaluating and analyzing alternatives, without much instruction. Furthermore, if asked, they can explain the used methods and obtained results quite well. In some cases, however, a seemingly convincing explanation may fall apart, revealing logical and computational errors.

Considering the MCDM process as a whole, the performance of LLMs is not as favorable. In subsequent MCDM stages, LLMs tend to "change their mind" without notice, modifying the already established model components: attributes, value scales, and decision rules. Consequently, this requires a lot of attention from the user's side, who has to check the outputs and perpetually remind the LLMs to remain consistent. This distracts the process and often carries the user away of the main decision-making task. Also, we should warn that in the preference modelling stage (section 5), LLMs suggest generalized decision preferences that might substantially differ from the user's subjective preferences, which need to be enforced explicitly.

In summary, LLMs can substantially contribute to the definition of attributes and alternatives, but are unsuitable for carrying out the whole MCDM process due to possible inconsistent and erroneous executions of the MCDM method. We believe that, given the current state of LLM development, it is more convenient and safer to use specialized and trusted MCDM software, such as DEXiWin. Nevertheless, LLMs evolve fast and we may expect substantial improvements in the future.

Acknowledgments

The authors acknowledge the financial support from the Slovenian Research and Innovation Agency for the programme Knowledge Technologies (research core funding No. P2-0103 and P5-0018).

References

[1] Kulkarni, A.J. (Ed.), 2022: Multiple Criteria Decision Making. Studies in Systems, Decision and Control 407, Singapore: Springer, doi: 10.1007/978-981-16-7414-3_3.

[2] Kamath, U., Keenan, K., Somers, G., Sorenson, S., 2024: *Large Language Models: A Deep Dive: Bridging Theory and Practice*. Springer, 506p, ISBN-13 978-3031656460.

[3] Bohanec, M., 2022: DEX (Decision EXpert): A qualitative hierarchical multi-criteria method. In: *Multiple Criteria Decision Making* (ed. Kulkarni, A.J.), Studies in Systems, Decision and Control 407, Singapore: Springer, doi: 10.1007/978-981-16-7414-3_3, 39–78.

[4] Bohanec, M., Rajkovič, V., Bratko, I., Zupan, B., Žnidaršič, M., 2013: DEX methodology: Three decades of qualitative multi-attribute modelling. *Informatica* 37, 49–54.

[5] Ishizaka, A., Nemery, P., 2013: Multi-criteria decision analysis: Methods and software. Chichester: Wiley.

[6] Bohanec, M., 2024: *DEXiWin: DEX Decision Modeling Software, User's Manual, Version 1.2*. Ljubljana: Institut Jožef Stefan, Delovno poročilo IJS DP-14747. Accessible from https://dex.ijs.si/dexisuite/dexiwin.html.

[7] Banerjee, S., Agarwal, A., Singla, S., 2024. LLMs will always hallucinate, and we need to live with this. 10.48550/arXiv.2409.05746.

Landscape-Aware Selection of Constraint Handling Techniques in Multiobjective Optimisation

Jordan N. Cork
Andrejaana Andova
Jožef Stefan Institute and
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
{jordan.cork,andrejaana.andova}@ijs.si

Pavel Krömer
Technical University of Ostrava
Ostrava, Czech Republic
pavel.kromer@vsb.cz

Tea Tušar
Bogdan Filipič
Jožef Stefan Institute and
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
{tea.tusar,bogdan.filipic}@ijs.si

Abstract

Constrained multiobjective optimisation problems (CMOPs) are common in real-world optimisation. They often involve expensive solution evaluations and, therefore, it is helpful to know the best methods to solve them prior to actually solving them. These problems also tend to be relatively difficult for algorithms compared to the majority of test problems. This difficulty often presents itself in the infeasible region, calling for a focus on the constraint handling technique (CHT). The purpose of this work is to select the best CHT for problems with difficult constraint functions. This first involves the collection of a set of such problems. CHT selection is then conducted using problem characterisation and machine learning. The outcomes are positive in that prediction achieved a high accuracy. Additionally, further insights are provided into the features that describe CMOPs.

Keywords

constrained multiobjective optimisation, algorithm selection, problem selection, constraint handling techniques

1 Introduction

Real-world optimisation problems very often have multiple objectives and are subject to one or more constraints. This is the domain of constrained multiobjective optimisation (CMO). These problems are generally demanding to solve and have restrictions to the available computational budget. These restrictions make it all the more important to know the best method for solving the problem prior to actually attempting to solve it. This calls for an algorithm selection methodology.

One approach to algorithm selection, known as landscape-aware selection, is to first characterise the problem before conducting the algorithm run [2]. Characterisation involves the calculation of features used to describe the objectives and constraints, as well as their interaction. This is done using a small set of sampled solutions. Once the problem is characterised, knowledge of similar problems can be used to determine the best approach to solving it. This approach is taken in this study and applied to constraint handling techniques (CHTs). CHTs are methods designed to guide optimisation algorithms in dealing with infeasible solutions, by taking as input the problem constraints and candidate solutions, and producing outputs that either repair, penalize, or rank these solutions to balance feasibility with optimality.

There are three primary contributions from this work, all within the CMO domain. The first is related to the set of problems used to train the algorithm selection model. Real-world optimisation problems are often difficult to solve, particularly when they include constraints. The field requires a methodology for selecting a subset of problems with difficult constraint functions from the larger set of known problems. This is the first contribution. The CHT selection methodology is then tested on these problems. This methodology is the second contribution. Here, problem characterisation and machine learning are used to predict the best-performing CHT. The final contribution is a set of insights into the features used. The decision tree output by the CHT selection methodology provides significant insights into both which features are useful and what the features reveal about the problems.

The paper is further structured as follows. In Section 2, CMO is introduced, providing the required background. Section 3 describes the two selection methodologies, as well as the validation method used. Section 4 presents the experimental setup. In Section 5, the results from the experiments are presented. Finally, in Section 6, the work is summarised and future work is outlined.

2 Constrained Multiobjective Optimisation

Constrained multiobjective optimisation (CMO) involves the optimisation of two or more objective functions given one or more constraint functions. The constraints may be of the equality or inequality forms, however, in this study, only inequality constraints are considered. Such a CMO problem (CMOP) may be formulated as follows:

$$\begin{aligned} \text{minimize} \quad & f_m(\mathbf{x}), \quad m = 1, \ldots, M, \\ \text{subject to} \quad & g_j(\mathbf{x}) \le 0, \quad j = 1, \ldots, J, \end{aligned} \tag{1}$$

where $\mathbf{x} = (x_1, \ldots, x_D) \in \mathbb{R}^D$ is a D dimensional *solution vector*, $f_m(\mathbf{x})$ are the *objective functions*, and $g_j(\mathbf{x})$ the inequality *constraint functions*. M is the number of objectives and J the number of inequality constraints.

CMO requires an indicator for assessing the quality of the set of optimal points. This indicator is I^{CMOP}. It was proposed in [19] to handle quality assessment in the three following situations. When no feasible solutions are found, it uses the minimum constraint violation. When feasible solutions are found, but these are outside of the region of interest (ROI) bound by the given reference point (RP), the distance to the ROI is used. Finally, when solutions are found within the ROI, it uses the hypervolume (HV). The HV measures the portion of the objective space dominated by the set of solutions relative to the RP. $I^{\text{CMOP}}$was proposed as a value to be minimised. However, it is commonly maximised based on the `moarchiving` package implementation [9]. On top of I^{CMOP}, the maximised area under the runtime profile curve is used to

Information Society 2025, Ljubljana, Slovenia
© 2025 Copyright held by the owner/author(s).
https://doi.org/10.70314/is.2025.skui.3280

measure the anytime performance of the algorithm [8]. Here, the runtime profile is the proportion of performance targets attained with respect to the evaluation number.

Many methodologies in CMO use an I^{CMOP}value with normalised function values. For this, the function values of the problems' optimal solution set are required. Together, these are known as the Pareto front. The Pareto front may be obtained empirically, through knowledge of the problems construction. Often this is not possible, however, and, therefore, algorithm runs are used to construct an approximation of the front.

In [4], there are 13 benchmark suites listed, consisting of 139 test problems. These test problems can be instantiated in various numbers of dimensions and objectives. This then allows for a substantially larger number of test problem instances to be generated based on these 139 base test problems.

Problem characterisation is conducted using exploratory landscape analysis (ELA) features [16]. Work done in [1] has listed 80 such features for CMO. These come from three landscapes: the multiobjective, violation and multiobjective-violation landscapes. The features can be computed via sampling or random walks.

There are several constraint handling techniques. Four of these are considered in our study. The first is the constrained-domination principle (CDP), proposed along with the NSGA-II algorithm [5]. This is a feasibility first approach, where feasible solutions are preferred over infeasible ones. The penalty CHT is a classic method and applies a penalty value to the objective values [20], either statically or dynamically. The Improved-Epsilon (I-Epsilon) CHT was designed to work with the MOEA/D algorithm [7]. It dynamically adjusts the ϵ value based on the number of feasible solutions. Solutions are considered feasible if they are less than the ϵ value. Finally, stochastic ranking (SR) uses a probability value to switch between comparing solutions based on objectives or constraints [18].

3 Methodology

This section presents the methodologies used in the study. First, the methodology for selecting the hard test problems is presented, followed by the methodology for selecting the appropriate CHT and the means for testing the model.

3.1 Difficult Problem Selection

Testing the CHT selection methodology requires test problems. Test problems with too easy constraint functions are less likely to show differences among the CHTs, as algorithms will spend less time dealing with infeasible solutions. More difficult constraint functions, on the other hand, will force the algorithm to deal with infeasible solutions longer and, therefore, give the CHTs time to show their differences. Test problems with difficult constraint functions are then desired for our testing.

As mentioned in Section 2, anytime performance is measured using the area under the runtime profile curve (AUC), with the maximised I^{CMOP}as the indicator. In this study, difficulty is determined based on the anytime performance of a set of algorithms, $\mathcal{A}$. Each of the algorithms is run on the problem R times and the average AUC is taken. This is to ensure robustness. It should be noted that when recording the runs, an archive of all non-dominated solutions is kept and the I^{CMOP}value from this archive is recorded at each solution evaluation. The budget must also be chosen, with budgets allowing algorithm convergence preferred. The maximum average AUC is then used as the problem difficulty, with lower values signifying harder problems. This is formulated as follows:

$$\text{Difficulty}(p) \;=\; 1 - \max_{a \in \mathcal{A}} \left(\frac{1}{R} \sum_{r=1}^{R} \text{AUC}(p, a, r) \right) \qquad (2)$$

This problem difficulty is calculated for each of the problems in the set of problems, $\mathcal{P}$.

Within the current selection, there will still be cases where all CHTs perform roughly the same on the problem. These problems are removed using statistical and practical threshold tests on the final I^{CMOP}values from the 30 runs. Given a normal distribution cannot be ensured in the 30 values from each of the algorithm runs, the Kruskal-Wallis test is used [11]. It determines if independent samples come from the same distribution. However, this still leaves problems with no practical differences in their scores. To filter these out, the mean scores are tested for if they vary more or less than a small delta and those that vary less are removed.

Following the filtering out of problems where no meaningful differences are observed, the $\mathcal{N}$ most difficult problems from the remaining set are selected. This leaves one with a suite of difficult problems upon which at least one of the algorithms from $\mathcal{A}$ performs differently.

3.2 Constraint Handling Technique Selection

The general concept for CHT selection is as follows. First, a machine learning model is trained using the features from each problem in the training set. The labels are the best-performing CHTs on each problem. At inference time, features are calculated on the problem in question (note: this consumes a portion of the available budget). These features are used as input to the machine learning algorithm. The resulting model then predicts the best-performing CHT for use during the run.

Each step will now be described in more detail. The first step is to choose a base algorithm and a set of algorithm-relevant CHTs. The preferred approach would be to select the most appropriate algorithm for the problem to be solved at inference time.

The second step is generating the training data for the machine learning model. First, the features for each of the problems in the training set are gathered. The labels must then be computed, which requires algorithm runs; 30 for each CHT. For this, the budget must be selected carefully. The model, at inference, can be expected to work well only if the budget is the same as it was in training. The average final values from the 30 runs are then taken for each CHT. In CMO, these are the average final I^{CMOP}values, which are being maximised. The CHT with the highest value is then selected as the best-performing CHT. This is used as the label. Once this has been done for each of the problems in the training set, the training data is complete.

The third step is to train the model. A decision tree is preferred for its explainability properties. To enhance the explainability of the model, the depth of the tree should be kept at a minimum. Testing is described in the next subsection. Once complete, i.e. trained with all training data, the model is available for inference.

3.3 Cross-Validation Testing

Testing the model involves a leave-one-problem-out cross-validation approach. Here, a problem is taken out of the training set and left as the test problem. The model is then trained on the data from the remaining problems in the training set. To predict the best-performing CHT, the features from the test problem are used as input to the model. The model then makes a prediction for the best-performing CHT. This is compared to the actual result.

The methodology makes allowances for when two or more CHTs perform similarly well on the same problem. The prediction made by the algorithm is then correct if it selects any of these. Determining if two or more CHTs are statistically the same is achieved through the use of a statistical test, which in this case was the Mann-Whitney U test [15]. Again, this test was chosen because a normal distribution cannot be ensured in the resulting final values from the runs. The process is as follows. The CHT with the best mean score is selected, then each of the other CHTs are tested individually against the best-performing CHT to determine if they are equivalent, forming the set of best-performing CHTs. If the predicted label is within this set, it is considered correct. This process is conducted for all problems in the training set and a final percentage of correct predictions is given.

4 Experimental Setup

In this section, the inputs to the methodologies are described, along with the packages used throughout.

There are several inputs to the difficult problem selection methodology. First, there is the set of problems, $\mathcal{P}$. The dimensions chosen were 2, 3, 5, 10 and 30, with only biobjective problems considered. This resulted in 375 problem instances. The problems were translated from Matlab by hand or taken from pymoo [3].

For $\mathcal{A}$, i.e. the set of algorithms, the natural choice was to choose a base algorithm with different constraint handling techniques. The base algorithm chosen was NSGA-II [5]. This was used for its versatility with regards to adding various CHTs. Regarding CHTs, CDP, penalty, I-Epsilon and SR were chosen for their compatibility with NSGA-II. CDP was provided as default with NSGA-II by pymoo. The others were implemented by hand. The penalty value selected was a static 100, while the settings for all others were the proposed defaults. R was set at 30.

The number of difficult problems selected, $\mathcal{N}$, was set at 20. This number is adequate to test the methodology while still being small enough to manage. The budget selected was the one to be used throughout the study, i.e. $10{,}000 \cdot D$. The delta value for detecting practical differences was set at 0.001.

For the CHT selection methodology, the choice of training problems was the set of difficult problems derived from the setup above. The base algorithm and CHTs were the same as those selected above. The model selected was a decision tree (scikit-learn [17]). The tree depth parameter was the only parameter tuned. This tuning was done manually, decreasing from 10 to 3, until the performance began to reduce. Finally, the problem features used were the 80 features described in [1]. These were calculated with a sample size of $1{,}000 \cdot D$. The random walks were simulated using these same samples.

5 Results

In this section, the results from carrying out the methodologies are described. First, the construction of the set of difficult problems is discussed. Then, the experimental results are presented. Finally, the resulting decision tree is discussed.

The difficulty of each problem was calculated as described in Section 3. The results were heavily skewed towards the easy problem side. With the $\mathcal{N}$ parameter set to 20, that many problems were selected. The difficulties of these ranged from 0.202 to 0.976. The selected problems are listed in Table 1 in order of descending difficulty. They include 5, 10 and 30 dimensional problems, with 2 and 3 dimensional problems clearly being easier to solve.

Table 1: The results from cross-validation testing using the leave-one-problem out methodology. The first column lists the test problems in order of difficulty (descending). D indicates the dimensionality. All problems are biobjective. The models were trained on all problems in the list, bar the test problem in question. 'Actual' lists the best-performing CHT labels, while the prediction column shows the predicted label. If the predicted label is in the actual labels list, the prediction is considered correct. The CHT labels 0, 1, 2 and 3 are CDP, penalty, I-Epsilon and SR, respectively.

Problem	D	Diffic.	Pred.	Actual	Correct
DC2-DTLZ3	30	0.976	2	[2]	Yes
DC2-DTLZ1	30	0.965	2	[2]	Yes
DC2-DTLZ1	10	0.541	2	[2]	Yes
DC2-DTLZ3	10	0.528	2	[2]	Yes
NCTP7	30	0.489	0	[0, 3]	Yes
NCTP8	10	0.355	3	[0, 1, 3]	Yes
NCTP15	10	0.339	3	[0, 1, 3]	Yes
DOC3	10	0.330	1	[0, 1, 3]	Yes
NCTP2	10	0.284	3	[0, 1]	*No*
NCTP1	10	0.279	3	[0, 1, 3]	Yes
NCTP7	10	0.269	3	[0, 3]	Yes
CTP6	30	0.257	1	[0, 1, 2]	Yes
CTP8	30	0.249	0	[0, 1, 2]	Yes
C1-DTLZ3	30	0.240	2	[0, 1, 2]	Yes
DC2-DTLZ1	5	0.230	2	[2]	Yes
CTP8	10	0.227	0	[0, 1, 2]	Yes
DC2-DTLZ3	5	0.219	2	[2]	Yes
DC3-DTLZ1	30	0.214	2	[2]	Yes
NCTP17	10	0.203	0	[0, 1, 2]	Yes
NCTP10	10	0.202	1	[0, 1, 2]	Yes

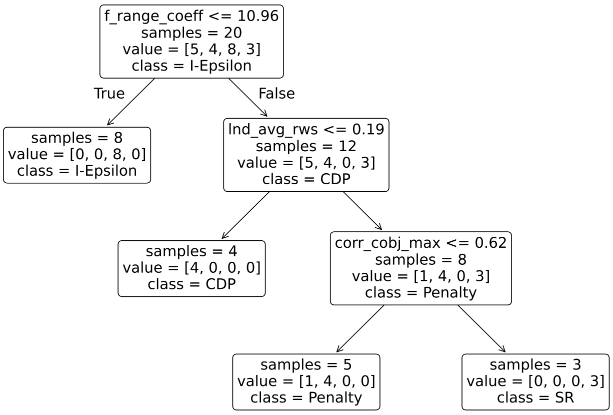


Figure 1: The decision tree built on all the training data. It is used to predict the four CHTs. The indices of the values in the value lists, indicating the number of instances, signify CDP, penalty, I-Epsilon and SR, respectively.

The problems come from the following suites: DC-DTLZ [13], NCTP [12], DOC [14], CTP [6] and C-DTLZ [10].

Table 1 additionally shows the results from the cross-validation testing phase of the experiments. As described in Section 3, each problem was given its turn as the test problem, while the others acted as training problems. For 95% of these, the model predicted correctly from the set of actual best-performing CHTs.

Figure 1 shows the decision tree that resulted from training on all of the available data. As it can be seen, the decision tree leaf nodes are nearly pure, meaning it achieved near 100% accuracy on the training data. Due to its high accuracy on the test data and the low tree depth, this is not believed to be overfit.

Only 3 of the 80 supplied features were included in the model, indicating their importance in identifying appropriate CHTs. The first of these, separating out I-Epsilon, was f_range_coeff (difference between the maximum and minimum of the absolute value of the linear regression model coefficients, where the model is fitted between the decision variables and the unconstrained ranks). This is a multiobjective landscape feature, focusing on variable scaling. The second feature, separating out CDP, was lnd_avg_rws (average proportion of locally non-dominated solutions in the neighbourhood). This is a multiobjective-violation landscape feature, focusing on evolvability, i.e. the degree to which the problem landscape facilitates evolutionary improvement. The final feature, distinguishing between penalty and SR, was corr_cobj_max (the maximum of the constraints and objectives correlation). This is also a multiobjective-violation landscape feature, focusing on evolvability. It should be noted that the features are not all related to the violation landscape, but also deal with the objective functions.

6 Conclusion

In this study, the focus was on the needs of real-world CMOPs. These problems are often difficult for algorithms to solve and require expensive solution evaluations. Given the cost of these evaluations, it is helpful to know the best method for solving the problem prior to actually solving it. To address this, the study focused on selecting the most appropriate CHT, a crucial component of any algorithm operating in CMO. For this selection task, it was critical to test on problems with difficult constraint functions. These problems elicit the most variation among CHTs.

The proposition was made for a methodology that selects problems with difficult constraint functions from a larger set, with the end goal of conducting CHT selection. This methodology involved first collecting a large set of CMOPs, then running a set of algorithms on them to determine their difficulty. Problems that were easy to solve or showed no variation in algorithm performance were discarded, as they provide no value in future CHT selection tasks. The methodology finally produced a set of $\mathcal{N}$ problems.

This set of difficult problems was used in the second methodology proposed, i.e. selecting CHTs using problem characterisation and machine learning. Four CHTs were chosen and added to the NSGA-II algorithm. These were CDP, penalty, I-Epsilon and SR. The goal of the selection task was to select the best-performing CHT on a given problem, noting that several CHTs can perform best. The methodology was evaluated using cross-validation, with the leave-one-problem-out method. The findings from testing were positive and indicate that it is possible to select the most appropriate CHT for a given difficult problem. Further, the final decision tree trained on all the considered difficult problems provides insights into the features characterising CMOPs.

In future work, the plans are to extend the CHT selection methodology to the broader domain of algorithm selection.

Acknowledgements

The authors acknowledge the financial support from the Slovenian Research and Innovation Agency (research core funding No. P2-0209 “Artificial Intelligence and Intelligent Systems”, and projects No. N2-0254 “Constrained Multiobjective Optimization Based on Problem Landscape Analysis” and GC-0001 “Artificial Intelligence for Science”).

References

[1] Hanan Alsouly, Michael Kirley, and Mario Andrés Muñoz. 2023. An instance space analysis of constrained multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 27, 5, 1427–1439. DOI: 10.1109/TEVC.2022.3208595.

[2] Andrejaana Andova, Jordan N. Cork, Aljoša Vodopija, Tea Tušar, and Bogdan Filipič. 2024. Predicting algorithm performance in constrained multiobjective optimization: A tough nut to crack. In *Applications of Evolutionary Computation*. Stephen Smith, João Correia, and Christian Cintrano, editors. Springer Nature Switzerland, Cham, 310–325. DOI: 10.1007/978-3-031-56855-8_19.

[3] Julian Blank and Kalyanmoy Deb. 2020. Pymoo: Multi-objective optimization in Python. *IEEE Access*, 8, 89497–89509. DOI: 10.1109/ACCESS.2020.2990567.

[4] Jordan N. Cork and Bogdan Filipič. 2025. A Bayesian optimization approach to algorithm parameter tuning in constrained multiobjective optimization. In *Optimization and Learning*. Bernabé Dorronsoro, Martin Zagar, and El-Ghazali Talbi, editors. Springer Nature Switzerland, Cham, 109–122. DOI: 10.1007/978-3-031-77941-1_9.

[5] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6, 2, 182–197. DOI: 10.1109/4235.996017.

[6] Kalyanmoy Deb, Amrit Pratap, and T. Meyarivan. 2001. Constrained test problems for multi-objective evolutionary optimization. In *Proceedings of the First International Conference on Evolutionary Multi-Criterion Optimization, EMO 2001*. Springer, 284–298. DOI: 10.1007/3-540-44719-9_20.

[7] Zhun Fan, Wenji Li, Xinye Cai, Han Huang, Yi Fang, Yugen You, Jiajie Mo, Caimin Wei, and Erik Goodman. 2019. An improved epsilon constraint-handling method in MOEA/D for CMOPs with large infeasible regions. *Soft Computing*, 23, 12491–12510. DOI: 10.1007/s00500-019-03794-x.

[8] Nikolaus Hansen, Anne Auger, Dimo Brockhoff, and Tea Tušar. 2022. Anytime performance assessment in blackbox optimization benchmarking. *IEEE Transactions on Evolutionary Computation*, 26, 6, 1293–1305. DOI: 10.1109/TEVC.2022.3210897.

[9] Nikolaus Hansen, Nace Sever, Mila Nedić, and Tea Tušar. 2024. Moarchiving: Multiobjective nondominated archive classes with up to four objectives. https://github.com/CMA-ES/moarchiving. GitHub repository. (2024).

[10] Himanshu Jain and Kalyanmoy Deb. 2014. 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. DOI: 10.1109/TEVC.2013.2281534.

[11] William H. Kruskal and W. Allen Wallis. 1952. Use of ranks in one-criterion variance analysis. *Journal of the American Statistical Association*, 47, 260, 583–621. DOI: 10.1080/01621459.1952.10483441.

[12] Jia-Peng Li, Yong Wang, Shengxiang Yang, and Zixing Cai. 2016. A comparative study of constraint-handling techniques in evolutionary constrained multiobjective optimization. In *2016 IEEE Congress on Evolutionary Computation (CEC)*, 4175–4182. DOI: 10.1109/CEC.2016.7744320.

[13] Ke Li, Renzhi Chen, Guangtao Fu, and Xin Yao. 2019. Two-archive evolutionary algorithm for constrained multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 23, 2, 303–315. DOI: 10.1109/TEVC.2018.2855411.

[14] Zhi-Zhong Liu and Yong Wang. 2019. Handling constrained multiobjective optimization problems with constraints in both the decision and objective spaces. *IEEE Transactions on Evolutionary Computation*, 23, 5, 870–884. DOI: 10.1109/TEVC.2019.2894743.

[15] Henry B. Mann and Donald R. Whitney. 1947. On a test of whether one of two random variables is stochastically larger than the other. *The Annals of Mathematical Statistics*, 18, 1, 50–60. DOI: 10.1214/aoms/1177730491.

[16] Olaf Mersmann, Bernd Bischl, Heike Trautmann, Mike Preuss, Claus Weihs, and Günter Rudolph. 2011. Exploratory landscape analysis. In *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*, 829–836. DOI: 10.1145/2001576.2001690.

[17] Fabian Pedregosa et al. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 85, 2825–2830. https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf.

[18] Thomas P. Runarsson and Xin Yao. 2000. Stochastic ranking for constrained evolutionary optimization. *IEEE Transactions on Evolutionary Computation*, 4, 3, 284–294. DOI: 10.1109/4235.873238.

[19] Aljoša Vodopija, Tea Tušar, and Bogdan Filipič. 2025. Characterization of constrained continuous multiobjective optimization problems: A performance space perspective. *IEEE Transactions on Evolutionary Computation*, 29, 1, 275–285. DOI: 10.1109/TEVC.2024.3366659.

[20] Yonas Gebre Woldesenbet, Gary G. Yen, and Biruk G. Tessema. 2009. Constraint handling in multiobjective evolutionary optimization. *IEEE Transactions on Evolutionary Computation*, 13, 3, 514–525. DOI: 10.1109/TEVC.2008.2009032.

Explaining Deep Reinforcement Learning Policy in Distribution Network Control

Blaž Dobravec
Elektro Gorenjska d.d.
Kranj, Slovenia
blaz.dobravec@elektro-gorenjska.si

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

Abstract

In safety-critical settings – such as low-voltage electrical distribution networks – Deep Reinforcement Learning (DRL) policies are hard to deploy due to limited capability to explain why a particular sequence of actions is taken. We use Scenario-Based eXplainability (SBX) with temporal prototypes to explain the policy of our DRL agent. SBX clusters short time-windows of latent trajectories and uses their medoid trajectories as human-friendly summaries. Temporal prototypes map the embeddings of these medoids to actions, and generate explanations of the form "This scenario is similar to prototype $X \Rightarrow$ Do action Y." We apply our approach to a real low-voltage distribution network Srakovlje. Preliminary results show that our method offers practically useful human-friendly explanations for sequential decision making.

Keywords

deep reinforcement learning, explainability, voltage control, low-voltage distribution network, prototypes

1 Introduction

A rapid growth of renewable energy resources and a significant increase in electricity demand due to the electrification of transport and heating [8] are reshaping generation (e.g. distributed photovoltaic systems) and consumption (e.g. heat-pumps, electrical vehicles) in electrical distribution networks. Increasing reverse power flows and voltage variability in low-voltage networks strongly affect voltage profiles and make the network operation and control more challenging.

Deep reinforcement learning (DRL) has recently emerged as a powerful paradigm for sequential decision-making in complex, high-dimensional environments, with notable successes in games (Chess [18], Go [19], Atari [13]), autonomous driving [10], and industrial robotic process automation [7]. Voltage control in distribution networks shares similar characteristics, which makes DRL a promising methodology to solve the control problems in low-voltage networks.

While voltage control is standard at higher voltage levels (e.g., with STATCOMs), most LV research has focused on optimizing individual assets at the customer level [11, 6]. Recent comparisons indicate that DRL can outperform classical algorithms for micro-grid management with demand-side flexibility [14]. For instance, dueling double DQN (D3QN) has been used to reduce overvoltages in PV-rich networks [16]; model-free RL has optimized battery charging/discharging to increase self-sufficiency [12]; and effective consumption/generation strategies have been learned under price signals and network constraints [2, 1]. Given the growing heterogeneity of LV networks and the rise of behind-the-meter actuators, DRL methods are typically developed and validated first in simulation [4]. Their adoption and implementation are often hindered by a lack of explainability of these models.

We present a prototype-based explainability approach for DRL-based voltage control in LV distribution networks that directly exploits flexibility from prosumers. In our approach, the agent acts on the network's operating state, coordinating different flexibility options (e.g. photovoltaic systems, batteries, EVs, heat pumps). We focus on improving power quality by reducing voltage violations. Additionally, we use prototype based explainability to provide interpretation and reasoning behind the action.

2 Related Work

Explainable Artificial Intelligence (XAI) aims to make the decisions of models understandable to humans. The explanation process and the final result should be focused on generating explanations that are intuitive to us. Prototype-based explanations provide a compelling choice that is interpretable by design. XRL remains an active area of research. One such widely employed explainability technique, primarily used in image classification, is the *saliency map*, which bases its explanations on pixel-wise feature attribution [20]. Building on this idea, Sequeira et al. [17] made the agent's interactions with the environment the focal point of their *Interestingness Framework*.

In supervised learning, prototype networks explain predictions via similarity to learned or human-selected exemplars [3, 15]. Extending this paradigm to reinforcement learning, prototype-wrapper policies force decisions to be mediated by human-friendly prototypes (single state-snapshot); a recent example is the Prototype-Wrapper Network (PW-Net), which wraps a pre-trained agent and maps latent states to action decisions through prototype similarities [9]. Beyond interpretability, prototypes have been leveraged to improve representation learning and exploration efficiency: Proto-RL pre-trains prototypical embeddings and uses prototype-driven intrinsic motivation to accelerate downstream policy learning in pixel-based control [23]. In model-based RL, prototypical context learning has also been explored for dynamics generalization [22].

Despite the critical role of explainability in voltage control in low-voltage power systems, there is little research addressing this challenge. Zhang et al. [24] applied the SHAP explainability method to a deep reinforcement learning model for implementing proportional load shedding during under-voltage situations. They also used Deep-SHAP [25] to enhance the computational efficiency of their XAI model. The model's output elucidates its

Information Society 2025, Ljubljana, Slovenia

https://doi.org/10.70314/is.2025.skui.9459

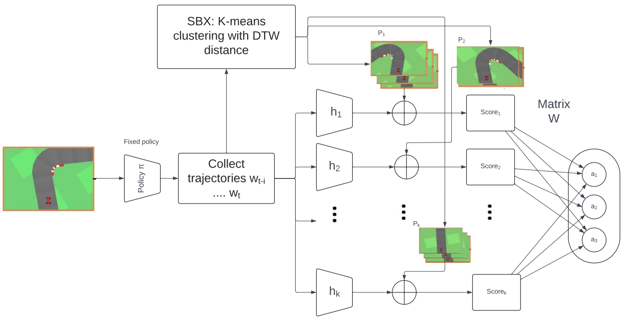

SBX: K-means clustering with DTW distance
P1
P2
Pk
Fixed policy
Policy π
Collect trajectories w_{t-i} .... w_t
h1
h2
hk
Score1
Score2
Scorek
Matrix W
a1
a2
a3

Figure 1: High-level SGTP pipeline: (1) collect latent windows; (2) SBX clustering and medoid selection; (3) train temporal-prototype layer; (4) case-based explanations during rollout.

predictions through a visualization layer and a feature importance layer that addresses both global and local explanations.

Existing research on explainability in power systems, particularly regarding voltage control, focuses on post-hoc explainability techniques. Compared to explanations for a single feature (individual voltage value) such as SHAP, our method considers the temporal component in the explanation process. To the best of our knowledge, this approach has not been applied to the explainability of the reinforcement learning field in this specific manner before.

3 SBX-guided Prototype Selection

We employ Scenario-Based eXplainability (SBX [5]) as an extension of the PWNet [9] to temporal prototypes (**prototypes of trajectories, not just snapshots of the state space**) to provide global, scenario-level structure and local, time-resolved explanations for a trained control policy. SBX is used to partition behavior and select representative temporal prototypes. On top of the SBX-selected prototypes (without any human-defined prototypes), we train a temporal prototype model that maps latent features to actions. This yields a two-tier view: SBX provides a summary of behavior, while temporal prototypes expose time-local patterns and their nearest neighbors that drive actions.

3.1 Data Preparation and Latent Extraction

We consider a trained policy π acting in discrete time. A trajectory is a sequence of observation–action pairs. For analysis, we operate on fixed-length trajectories of length L:

$$w_t = \big((o_t, a_t), \dots, (o_{t+L-1}, a_{t+L-1})\big), \quad t = 0, \dots, T - L.$$

Observations are first mapped by the frozen policy backbone to latent vectors $x_t \in \mathbb{R}^d$. We denote the latent trajectory by $X_t = (x_t, \dots, x_{t+L-1}) \in \mathbb{R}^{L \times d}$. We collect an offline dataset by rolling out the trained PPO agent and recording, at each time step, the policy's penultimate-layer latent vector and the corresponding environment action. This yields per-episode sequences of latents and actions which are then converted into trajectories of length L. The supervised target for each trajectory is the action at its last real-time step.

3.2 SBX Prototype Selection

SBX is performed in the latent space by clustering window embeddings with k-means over a range of cluster counts and selecting the number of clusters via a silhouette-style score. Within each selected cluster, the medoids (nearest to the centroids) are taken as temporal prototypes. Optionally, flattened action windows are concatenated to latent trajectories before k-means to bias prototype selection toward action-discriminative regions. The SBX step produces a prototype tensor of shape (K, L, d).

3.3 Temporal Prototype Model

We introduce K temporal prototypes $\{P_k\}_{k=1}^{K}$, each a length-L latent template $P_k \in \mathbb{R}^{L \times d}$ selected by SBX (medoids). A shared temporal encoder $g_\theta : \mathbb{R}^{L \times d} \to \mathbb{R}^p$ maps trajectories to embeddings $z_t = g_\theta(X_t)$ and prototypes to $e_k = g_\theta(P_k)$. Following PW-Net, prototype activations use an L2-to-activation mapping.

$$a_k(t) = \log \frac{\|z_t - e_k\|_2^2 + 1}{\|z_t - e_k\|_2^2 + \varepsilon}, \quad \varepsilon > 0. \tag{1}$$

Outputs are linear in activations, $y_t = W\,a(t)$, optionally post-processed to valid actions (Tanh/ReLu for steer/gas/brake). The schematics of the algorithm is outlined in Figure 1.

3.4 Inference and Explanations

At test time, we slide a window over trajectories, compute activations $a_k(t)$, and predict actions y_t. Explanations are provided by (i) the SBX scenario summaries (offline) and (ii) nearest-neighbor windows to each prototype in the encoder embedding space.

- **Scenario-level** (global): SBX clusters and medoids summarize typical behaviors.
- **Temporal prototype-level** (local): per-prototype nearest windows (and prototype self-windows) illustrate characteristic action trajectories.

For each time step, form the most recent latent window, compute the encoder embedding and prototype activations, map them linearly to actions, and apply Tanh/ReLu post-processing. Key hyperparameters are L (window length), encoder size p, and learning rate. We select them on a held-out set using validation MSE and qualitative visualization of nearest-neighbor trajectories.

4 Experiments

4.1 Simulation and voltage control policy

We examine a real-world low-voltage distribution network consisting of 26 consumers, of which 7 are active consumers. Those active consumers are equipped with small solar plants (11kWp). The total yearly consumption in this network is negative, meaning that the solar plants are producing more electricity than is needed. A visual representation of the network is displayed in Fig. 2.

The learning process extended over 1500 episodes, each containing 96 steps (representing a 15-minute interval across one day). We evaluated the model every 20 episodes (1 epoch). In this network, we focus on handling mainly high voltages as those are a bigger problem in our example.

4.2 Explaining a Simulation

We consider a real low-voltage distribution network. An observation/state is the vector of per-bus voltage magnitudes $s = [v_1, \dots, v_n]$ (in per unit). Actions are per-active-consumer flexibility commands $a = [\alpha_1, \dots, \alpha_m]$ with $\alpha_i \in [-1, 1]$: negative values decrease consumption (or increase net export) and positive values decrease the generation for active consumers (bounded by their instantaneous battery output). The agent acts every 15

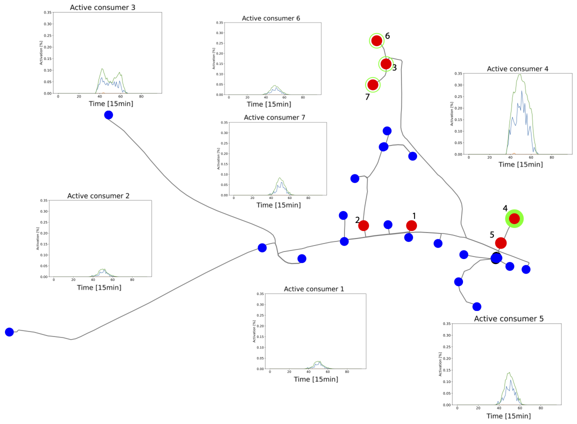


Figure 2: The network Srakovlje is located in Gorenjska region (north-western part of Slovenia). Active consumers (red), and their most representative activations are displayed with the corresponding graph. Green circles denote the most common over-voltage buses prior to voltage control. The width of the green circle indicates the severity of the original over-voltage measurements.

minutes; episodes comprise 96 steps (one day). The goal is to keep voltages within operating limits while minimizing interventions and losses.

Following prior work on distribution-voltage control [21], we use a reward that balances voltage quality, activation effort, and network losses. Trajectories are generated by a PPO policy trained in this environment.

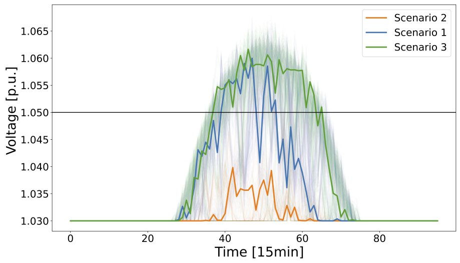


Figure 3: Centroids and underlying medoids of the scenarios in the Power Control environment. The individual color represents the average voltage signal in the network corresponding to the scenarios.

We used trajectories with length $L = 96$ which gives us $K = 3$ prototypes (Figure 3). Scenario selection via a silhouette-style criterion over $k \in \{2, \ldots, 8\}$ yielded a preferred $k = 3$ scenarios. Representative scenario-level activation summaries are shown in Figure 4. **Task fidelity:** offline action-level discrepancy against the reference policy (mean-squared error over held-out trajectories at the final step) was MSE = 3.218. **Scenario quality:** stored similarity scores by k were: $k = 2$: 0.131, $k = 3$: 0.118, $k = 4$: 0.083, $k = 5$: 0.082, $k = 6$: 0.089, $k = 7$: 0.093, $k = 8$: 0.096. A recomputed silhouette for the chosen $k = 3$ partition gave 0.099 with per-scenario supports $[4212, 7312, 5912]$ trajectories, indicating three regimes with substantial coverage. **Prototype locality:** In the latent space, the average distance from each prototype to its top-25 nearest trajectories was 0.124 on average, suggesting coherent time-local patterns.

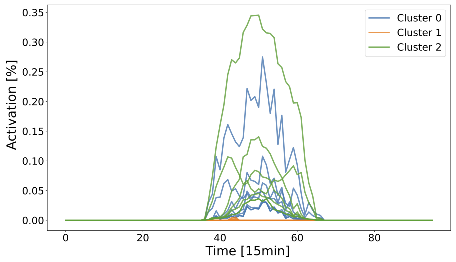


Figure 4: Representative prototypes in the Power Control environment. Each color represents the Scenario, and the individual line represents the activations by the individual active consumers.

4.3 Results

Fidelity. Across both domains, the prototype-based policy closely tracks the black-box in task reward, while achieving low action discrepancy on held-out episodes. This suggests that mediating actions through temporal prototypes does not materially degrade performance.

Global structure. SBX consistently discovers a small set of recurring scenarios that align with intuitive regimes (straight driving vs. cornering in continuous control; typical operating conditions in slower dynamics). Scenario summaries (state/action mean±std) are distinct and exhibit stable temporal patterns.

Local interpretability. For representative episodes, the nearest-neighbor aggregates around each prototype show coherent time-local patterns, and the most influential prototypes (largest contributions) align with observed actions. Explanations adopt a case-based form, relating current decisions to similar prototypical windows.

Performance Analysis. We compared the rewards across different policy architectures. Table **??** presents the results of running 20 episodes for each policy variant, measuring key performance metrics including mean reward, consistency (standard deviation), and coefficient of variation (CV) as a measure of reliability.

Over 20 episodes, the Base policy achieves the highest mean reward (221.8; range 201.0–257.5). PWNet closely matches the Base with a mean of 220.7 (≈ 0.5% lower; range 185.8–249.5), indicating that mediating decisions through prototypes incurs negligible performance loss. The Temporal PWNet trades some reward for interpretability, averaging 211.5 (≈ 4.7% below Base; range 168.4–231.8). Overall, relative performance is: Base ≈ 100%, PWNet ≈ 99%, Temporal PWNet ≈ 95%.

The results demonstrate several key insights about our approach. The Base Policy achieves the best rewards. The PWNet Policy shows comparable performance, indicating that prototype-based explanations can be achieved without significant performance degradation. Our Temporal PWNet + SBX approach achieves a mean reward of 211.47 ± 14.60, representing a modest performance trade-off in exchange for enhanced interpretability through temporal prototypes and scenario-guided explanations.

5 Discussion

This work introduces Scenario-Guided Temporal Prototypes, which combines global scenario discovery (SBX) with local, time-resolved prototypes to explain DRL decisions in voltage control problem in power networks. We observe that temporal prototypes can approximate black-box actions off-line with low discrepancy while forcing decisions through human-friendly exemplars. SBX discovers a small number of recurring regimes, with clear scenario-level summaries (Figure 3) and consistent prototype neighborhoods. This supports case-based reasoning over the policy's temporal dynamics rather than single-step feature attributions. Tight nearest-neighbor bands and balanced per-scenario support indicate that selected prototypes are representative rather than outliers.

The limitations of our current approach include reliance on a particular windowing choice and off-line evaluation that does not account for control feedback. Extremely imbalanced or highly non-stationary data may complicate selection. Prototype interpretability depends on the quality of medoids and the clarity of the associated concepts; domains lacking clear temporal motifs may benefit less from temporal prototypes and may also see degradation in performance. SBX does not identify the outliers that might be important for the agent to succeed. The identification of such states within the current architecture will be explored in future work. Future work also includes dynamic prototype lengths and human-in-the-loop curation tools for prototype editing and labeling.

6 Conclusion

We presented a pre-hoc interpretability framework that (i) discovers scenario structure from trajectories and (ii) explains actions via temporal prototypes. The approach yields faithful, time-resolved explanations without materially degrading control quality, as demonstrated in Power Network voltage control. Explanations take a case-based form—"this situation is similar to prototype X"—and are grounded by scenario summaries and prototype locality.

Beyond improving transparency, our approach offers practical steps: scenario coverage, per-scenario prototype counts, and nearest-neighbor coherence expose where explanations are strong or require refinement. Looking ahead, we plan to enable interactive prototype curation, incorporate uncertainty-aware explanation scores, and explore joint training schemes that couple prototype-based interpretability with context-aware latent dynamics. We will explore the sensitivity of the hyperparameter L to the actual training success. We have also identified that the fidelity metrics beyond the MSE will be necessary to explore. At this moment comparison to the saliency methods or SHAP explanations is still challenging due to the different nature of explanations (one being feature step-wise based and the other being multi-step and comparison based). Together, these steps can help bridge the gap between high-performing DRL policies and the trust required for their deployment.

Acknowledgements

This work was partially supported by the Slovenian Research Agency (ARIS), grant L2-4436: Deep Reinforcement Learning for optimization of LV distribution network operation with Integrated Flexibility in real-Time (DRIFT), and from the Slovenian Research Agency (ARIS) as member of the research program Artificial Intelligence and Intelligent Systems (Grant No. P2-0209).

References

[1] Shahab Bahrami, Yu Christine Chen, and Vincent W. S. Wong. 2021. Deep reinforcement learning for demand response in distribution networks. *IEEE Transactions on Smart Grid*, 12, 1496–1506.

[2] Di Cao, Junbo Zhao, Weihao Hu, Fei Ding, Nanpeng Yu, Qi Huang, and Zhe Chen. 2021. Model-free voltage control of active distribution system with PVs using surrogate model-based deep reinforcement learning. *Applied Energy*, 306, Part A, (Nov. 2021).

[3] Chaofan Chen, Oscar Li, Chaofan Tao, Alina Barnett, Cynthia Rudin, and Jonathan Su. 2019. This looks like that: deep learning for interpretable image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 8930–8939.

[4] Ruisheng Diao, Zhiwei Wang, Di Shi, Qianyun Chang, Jiajun Duan, and Xiaohu Zhang. 2019. Autonomous voltage control for grid operation using deep reinforcement learning. *CoRR*, abs/1904.10597. arXiv: 1904.10597.

[5] Blaž Dobravec and Jure Žabkar. 2024. Explaining voltage control decisions: a scenario-based approach in deep reinforcement learning. In *Foundations of Intelligent Systems*. Springer Nature Switzerland, Cham, 216–230. isbn: 978-3-031-62700-2.

[6] Samar Fatima, Verner Püvi, and Matti Lehtonen. 2020. Review on the PV hosting capacity in distribution networks. *Energies*, 13, 18.

[7] Natanael Gomes, Felipe Martins, José Lima, and Heinrich Wörtche. 2022. Reinforcement learning for collaborative robots pick-and-place applications: a case study. *Automation*, 3, (Mar. 2022).

[8] European Union Policy Iniciative. [n. d.] Growing consumption in the european markets. https://knowledge4policy.ec.europa.eu/growing-consumerism. Accessed: 2022-11-10. ().

[9] Eoin M. Kenny, Mycal Tucker, and Julie A. Shah. 2023. Towards interpretable deep reinforcement learning with human-friendly prototypes. In *ICLR*.

[10] Bangalore Ravi Kiran, Ibrahim Sobh, Victor Talpaert, Patrick Mannion, Ahmad A. Al Sallab, Senthil Kumar Yogamani, and Patrick Pérez. 2020. Deep reinforcement learning for autonomous driving: A survey. *CoRR*, abs/2002.00444. arXiv: 2002.00444.

[11] Wong Ling Ai, Vigna Ramachandaramurthy, Sara Walker, and Janaka Ekanayake. 2020. Optimal placement and sizing of battery energy storage system considering the duck curve phenomenon. *IEEE Access*, 8, (Jan. 2020), 197236–197248. doi: 10.1109/ACCESS.2020.3034349.

[12] Brida V. Mbuwir, Fred Spiessens, and Geert Deconinck. 2018. Self-learning agent for battery energy management in a residential microgrid. In *2018 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, 1–6.

[13] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin A. Riedmiller. 2013. Playing atari with deep reinforcement learning. *CoRR*, abs/1312.5602. arXiv: 1312.5602.

[14] Taha Nakabi and Pekka Toivanen. 2020. Deep reinforcement learning for energy management in a microgrid with flexible demand. *Sustainable Energy Grids and Networks*, 25, (Dec. 2020).

[15] Meike Nauta, Sander van Bree, and Christin Seifert. 2021. Neural prototype trees for interpretable fine-grained image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 14933–14943.

[16] Alvaro Rodriguez del Nozal, Esther Romero-Ramos, and Angel Luis Trigo-Garcia. 2019. Accurate assessment of decoupled oltc transformers to optimize the operation of low-voltage networks. *Energies*, 12, 11.

[17] Pedro Sequeira and Melinda T. Gervasio. 2019. Interestingness elements for explainable reinforcement learning: understanding agents' capabilities and limitations. *Artif. Intell.*, 288, 103367.

[18] David Silver et al. 2017. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *CoRR*, abs/1712.01815. arXiv: 1712.01815.

[19] David Silver et al. 2017. Mastering the game of go without human knowledge. *Nature*, 550, 354–359.

[20] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2013. Deep inside convolutional networks: visualising image classification models and saliency maps. *CoRR*, abs/1312.6034.

[21] Jianhong Wang, Wangkun Xu, Yunjie Gu, Wenbin Song, and Tim C. Green. 2021. Multi-agent reinforcement learning for active voltage control on power distribution networks. *CoRR*, abs/2110.14300. arXiv: 2110.14300.

[22] Junjie Wang, Qichao Zhang, Yao Mu, Dong Li, Dongbin Zhao, Yuzheng Zhuang, Ping Luo, Bin Wang, and Jianye Hao. 2024. Prototypical context-aware dynamics for generalization in visual control with model-based reinforcement learning. *IEEE Transactions on Industrial Informatics*, 20, 9, 10717–10727. doi: 10.1109/TII.2024.3396525.

[23] Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. 2021. Reinforcement learning with prototypical representations. In *Proceedings of the 38th International Conference on Machine Learning* (Proceedings of Machine Learning Research). PMLR. https://arxiv.org/abs/2102.11271.

[24] Ke Zhang, Peidong Xu, and Jun Zhang. 2020. Explainable ai in deep reinforcement learning models: a shap method applied in power system emergency control. In *2020 IEEE 4th Conference on Energy Internet and Energy System Integration (EI2)*, 711–716.

[25] Ke Zhang, Jun Zhang, Pei-Dong Xu, Tianlu Gao, and David Wenzhong Gao. 2022. Explainable ai in deep reinforcement learning models for power system emergency control. *IEEE Transactions on Computational Social Systems*, 9, 2, 419–427.

Leveraging AI in Melanoma Skin Cancer Diagnosis: Human Expertise vs. Machine Precision

Anna-Katharina Herke
Applied Artificial Intelligence
Alma Mater Europaea

Anna-
katharina.herke@almamater.si

Abstract

Whilst relatively uncommon compared to other skin cancers, melanoma is one of the most aggressive forms of this cancer. Given early and accurate detection, the condition can be treated successfully. Despite advancements in dermoscopy, diagnostic variability among dermatologists persists, often delaying treatment. This paper investigates the performance of a deep learning model based on ResNet-50 against human dermatologists in melanoma detection, highlighting synergies between AI and human diagnostics. Our findings indicate that AI can be as accurate or better than individual dermatologist performance in key metrics like sensitivity and specificity, and that a workflow focused on collaboration in the diagnostic process yields superior outcomes compared to either approach alone.

Keywords

Melanoma, skin cancer diagnosis, AI in cancer diagnosis, dermatology

1 Introduction

Globally, melanoma accounts for a disproportionate number of skin cancer-related deaths despite being less common than other skin cancers like basal and squamous cell carcinomas. In the United States alone, melanoma only accounts for one in 100 cases of skin cancer, while causing the majority of deaths from this type of cancer [31]. Early detection dramatically improves prognosis, with five-year survival rates exceeding 90% when melanoma is identified at an early stage [1]. However, diagnostic accuracy in dermatology remains highly variable, dependent on clinician experience, lesion characteristics, and access to dermoscopic tools.

This variability presents a significant diagnostic challenge. Studies have revealed that dermatologists may miss up to one in five (20%) cases of melanoma. There is also disagreement between professionals on lesion categorization [3, 4]. Artificial intelligence (AI), particularly deep learning algorithms trained on large dermoscopic datasets, has emerged as a potential equalizer, capable of achieving and possibly exceeding the classification accuracy of dermatologists [1, 2].

AI's ability to analyze complex visual patterns in skin lesions offers a novel solution to diagnostic gaps. However, questions remain regarding its performance in clinical settings, generalizability potential biases, and ethical implications [14, 15]. This study aims to compare the diagnostic performance of a ResNet-50-based AI model with that of board-certified dermatologists and explore synergistic diagnostic workflows. We place specific emphasis on aspects of dataset composition, prospective evaluation design, and clinical integration to expand on the findings of previous studies.

2 Research Questions

This paper will focus on and attempt to answer the following research questions:

1. How does the diagnostic accuracy of an AI model compare to that of human dermatologists?
2. Can AI-human collaboration enhance melanoma detection outcomes?
3. What are the ethical and practical considerations for AI integration in clinical dermatology?

3 Related Work

Early studies such as Esteva et al. [1] demonstrated the power of artificial intelligence in skin cancer diagnostics. The authors showed that deep convolutional neural networks (CNNs) could match the diagnostic performance of dermatologists in melanoma classification. Haenssle et al. [2] confirmed these findings in a controlled reader study. Similarly, Brinker et al. [4] found that a CNN outperformed 86% of participating dermatologists.

Information Society 2025, 6–10 October 2025, Ljubljana, Slovenia

https://doi.org/10.70314/is.2025.skui.8366

Recent research has shifted toward examining the potential of collaborations between humans and AI. Tschandl et al. [3] and Allen et al. [26] found that AI-assisted diagnosis improved the accuracy of clinician diagnosis alone. Navarrete-Dechent et al. [7] conducted a prospective trial showing how synergistic diagnosis combining dermatologists and AI tools improved diagnostic accuracy.

However, limitations persist. Most studies use retrospective or experimental setups lacking real-world clinical integration. Few address model bias, particularly regarding skin tone and underrepresented populations [14, 15, 33, 34]. Those could lead to false diagnoses. Continued reliance on HAM10000 and institutional datasets restricts generalizability of research findings.

In addition, the absence of real-world patient context such as patient history and a physical exam may cause clinicians to underestimate diagnostic complexity. Furthermore, adoption barriers among clinicians remain underexplored at the time of writing [27].

This submission seeks to fill these gaps with a prospective evaluation of AI-human performance and practical deployment considerations.

4 Methods

4.1 Data Acquisition and Preprocessing

Dermoscopic images were sourced from the commonly used HAM10000 dataset [13], supplemented by institutional image archives. Inclusion criteria comprised high-resolution dermoscopic images of histopathologically confirmed melanomas and benign nevi. Exclusion criteria included images with low resolution, artifacts, or incomplete metadata.

All images underwent standardized preprocessing procedures such as resizing to 224×224 pixels, normalization, and augmentation (flipping, rotation, and contrast adjustments) to enhance generalizability [21, 23].

4.2 AI Model Architecture

For this study, we utilized a ResNet-50 CNN pretrained on ImageNet, fine-tuned on the melanoma dataset. The model incorporated dropout regularization and cross-entropy loss optimization. Training was conducted on NVIDIA GPUs using a 70/15/15 train-validation-test split. This architecture and training paradigm has demonstrated high performance in skin lesion classification tasks and is widely adopted in dermatology AI literature [1, 4].

4.3 Human Cohort and Diagnostic Protocol

Twenty board-certified dermatologists with 5–25 years of clinical experience participated. We asked each participant to review 100 randomized images. Images were presented in isolation, blind to patient history and pathology. Diagnoses were binary (melanoma vs. benign). In a second round, participants reviewed the same images with AI output overlays.

This two-phase diagnostic design aligns with previous human-versus-AI studies, notably those by Haenssle et al. and Tschandl et al., which examined both solo and AI-assisted diagnostic conditions [2, 3, 7]. Randomization and blinding ensure impartial evaluation, a standard methodological feature in comparative diagnostic trials [5, 6].

4.4 Evaluation Metrics

Performance was measured using sensitivity, specificity, area under the ROC curve (AUC-ROC), and average diagnostic time per image. Inter-rater agreement was assessed using Fleiss' kappa.

5 Results

5.1 AI vs Human Diagnostic Performance

The AI model achieved an AUC-ROC result of 0.94, with 89% sensitivity and 85% specificity. Dermatologists averaged an AUC of 0.87, with 82% sensitivity and 83% specificity. Notably, a total of 75% (15 out of 20) dermatologists were outperformed by the AI in sensitivity [4].

We further analyzed inter-rater variability among clinicians using Fleiss' kappa statistics. Without AI assistance, Fleiss' kappa was 0.58 (moderate agreement). With AI assistance, kappa increased to 0.72 (substantial agreement), indicating improved consensus among readers.

This improvement in agreement supports the claim that AI support enhances diagnostic reliability and synergizes with human expertise.

Table 1: Inter-Rater Variability

Scenario	Fleiss' Kappa
Clinicians Alone	0.58
Clinicians + AI Assist	0.72

Source: research performed in the course of this study

5.2 AI-Human Synergy Analysis

When assisted by AI, dermatologist sensitivity improved to 91%, and specificity rose to 87%, surpassing both the solo AI and unassisted human performance. Average diagnostic time dropped from 22 seconds to 15 seconds per image [28].

Table 2: Visual Summary of Results

Diagnostic Modality	Sensitivity	Specificity	AUC-ROC	Avg Time/ Image
AI Alone	89%	85%	0.94	3 seconds
Dermatologists Alone	82%	83%	0.87	22 seconds
Dermatologists + AI	91%	87%	0.96	15 seconds

Source: research performed in the course of this study

6 Discussion

We were able to affirm previous findings that artificial intelligence has the capacity to match or outperform dermatologists in the detection of melanoma [1, 5]. Moreover, diagnostic synergy between human experts and AI enhances overall performance, aligning with findings from Tschandl et al. [3] and Navarrete-Dechent et al. [7].

6.1 Ethical Considerations and Bias Analysis

Despite strong results when combining clinician expertise with AI in melanoma detection, concerns persist. These concerns begin even before the algorithm is applied. AI models may have been subject to biased training data. In this context, underrepresentation of darker skin tones remains problematic [14, 15]. As a result, AI may exacerbate healthcare disparities [20], and there remains a need for inclusive datasets and algorithmic transparency [19] to address these challenges.

To strengthen our analysis of bias and inclusivity, we present a descriptive breakdown of our dataset by skin type (Fitzpatrick scale):

Table 3: Skin Type Breakdown

Fitzpatrick Skin Type	**Number of Cases**	**Percentage (%)**
I–II (Light)	500	40
III–IV (Medium)	500	40
V–VI (Dark)	250	20
Total: 1,250 Images		

Source: research performed in the course of this study

This distribution allows for more robust discussion of skin tone bias and ensures inclusiveness in our findings. We acknowledge that the representation of darker skin types (V–VI) remains limited and may impact generalizability. Future studies should prioritize dataset balance for equitable AI performance.

In collaborative settings, explainability remains another challenge, as clinicians may distrust opaque AI decisions that lack transparency. Incorporating interpretable AI frameworks and continuous feedback loops can help address these issues [21].

6.2 Integrating AI into Clinical Practice

Adoption hurdles include clinician skepticism, workflow integration, and regulatory uncertainty [27, 25]. Real-world implementation requires AI tools to function as second readers, supporting—not supplanting—clinicians [6, 22].

Regulatory guidance from the FDA (2022) emphasizes post-market monitoring, performance transparency, and adaptive learning constraints. Clinician training, robust validation, and clear liability frameworks are essential for safe deployment.

7 Conclusion

This study highlights the promise of AI-human collaboration in melanoma diagnosis. A fine-tuned ResNet-50 model achieved diagnostic accuracy comparable to board-certified dermatologists and improved performance when integrated into clinician workflows. While AI holds transformative potential, challenges around bias, explainability, and regulatory oversight must be addressed to ensure equitable, trustworthy deployment.

Future work should focus on prospective clinical trials, patient-facing applications, and interdisciplinary frameworks for human-AI co-diagnosis. A hybrid diagnostic model, leveraging AI's speed and consistency with human intuition and contextual awareness, represents the future of dermatological practice.

As diagnostic models develop, so will technology. Improvements in AI, such as federated learning and enhanced explainability methods will lead to improved trust and adoption in clinical settings.

Acknowledgments

The completion of this analysis on melanoma skin cancer was achieved through the insightful contributions of researchers in the field whose work was pivotal for this analysis. I am thankful for the academic community and institutions that provided access to research databases and journals, which were essential for the literature review.
I would also like to extend my gratitude to the peer reviewers and editors at the IS.IJS.SI Conference, especially Matjaz Gams, whose valuable feedback enhanced the quality of this paper.

References

[1] Esteva, A., et al. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639), 115–118.

[2] Haenssle, H.A., et al. (2018). Man against machine: Diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma image classification. *Annals of Oncology*, 29(8), 1836–1842.

[3] Tschandl, P., et al. (2020). Human–computer collaboration for skin cancer recognition. *Nature Medicine*, 26(8), 1229–1234.

[4] Brinker, T.J., et al. (2019). Deep learning outperformed 136 of 157 dermatologists in a head-to-head dermoscopic melanoma image classification task. *European Journal of Cancer*, 113, 47–54.

[5] Phillips, M., et al. (2019). Assessment of accuracy of an artificial intelligence algorithm to detect melanoma in images of skin lesions. *JAMA Network Open*, 2(10), e1913436.

[6] Marchetti, M.A., et al. (2020). Artificial intelligence as a second reader in melanoma screening. *Journal of the American Academy of Dermatology*, 83(1), 188–194.

[7] Navarrete-Dechent, C., et al. (2022). Human-AI synergy in melanoma diagnosis: A prospective clinical trial. *Journal of the American Academy of Dermatology*, 86(3), 567–575.

[8] Liu, Y., et al. (2021). Deep learning for melanoma detection: A systematic review. *Journal of Investigative Dermatology*, 141(12), 2835–2844.

[9] Fujisawa, Y., et al. (2019). Deep learning-based image analysis of melanocytic lesions: Current status and future prospects. *Frontiers in Medicine*, 6, 99.

[10] Codella, N.C.F., et al. (2018). Skin lesion analysis toward melanoma detection: ISIC 2017 Challenge. *IEEE ISBI*, 168–172.

[11] Sood, T., et al. (2021). AI in dermatology: Challenges and opportunities. *Journal of Medical Systems*, 45(7), 1–8.

[12] Han, S.S., et al. (2018). Classification of the malignancy of skin lesions using deep learning-based image analysis. *PLoS One*, 13(11), e0205820.

[13] Tschandl, P., et al. (2018). The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific Data*, 5, 180161.

[14] Groh, M., et al. (2021). Evaluating racial bias in AI skin cancer models. *NEJM AI*, 1(1), 1–10.

[15] Daneshjou, R., et al. (2022). Disparities in dermatology AI performance on a diverse patient population. *Science Translational Medicine*, 14(645), eabq6147.

[16] Kittler, H., et al. (2016). Diagnostic accuracy of an artificial intelligence–based device for the evaluation of pigmented skin lesions. *Lancet Oncology*, 17(12), 1785–1793.

[17] Hollon, T.C., et al. (2020). Machine learning identifies surgical margins in patients with melanoma using stimulated Raman histology. *Cancer Research*, 80(4), 664–673.

[18] Brinker, T.J., et al. (2020). Skin cancer classification using convolutional neural networks: Systematic review. *J Med Internet Res*, 22(10), e20736.

[19] Yogananda, C.G., et al. (2021). A Survey on Explainable AI for Skin Lesion Analysis. *Front Med*, 8, 777911.

[20] Adamson, A.S., Smith, A. (2018). Machine learning and healthcare disparities in dermatology. *JAMA Dermatol*, 154(11), 1247–1248.

[21] Ghosal, A., et al. (2021). Deep learning for melanoma detection: A comprehensive review. *Artificial Intelligence Review*, 54(8), 5783–5819.

[22] Topol, E.J. (2019). High-performance medicine: The convergence of human and artificial intelligence. *Nature Medicine*, 25, 44–56.

[23] Han, S.S., et al. (2022). Federated learning for melanoma detection across institutions. *Nature Communications*, 13(1), 1–10.

[24] Ud Din, N., et al. (2023). Artificial Intelligence for Melanoma Diagnosis: A Decade of Progress. *Cancers*, 15(3), 876.

[25] Wong, A., et al. (2022). Ethical challenges of AI in melanoma diagnosis. *Lancet Digital Health*, 4(3), e156–e165.

[26] Allen, J., et al. (2021). Human–Machine Collaboration in Skin Lesion Diagnosis. *JAMA Dermatol*, 157(8), 947–954.

[27] Jones, O.T., et al. (2021). Barriers to AI adoption in dermatology: A clinician survey. *British J Dermatol*, 185(2), 345–352.

[28] Yamada, M., et al. (2022). An AI tool helped reduce dermatologist diagnosis times and errors: A retrospective study. *Artificial Intelligence in Medicine*, 129, 102317.

[29] Udrea, A., et al. (2020). Accuracy of a smartphone application for triage of skin lesions based on machine learning in a primary care setting. *JAMA Network Open*, 3(6), e2036362.

[30] FDA. (2022). Regulatory considerations for AI/ML-based medical devices. FDA Guidance Document.

[31] American Cancer Society (2025). Key Statistics for Melanoma Skin Cancer. Accessed on May 26, 2025 under: https://www.cancer.org/cancer/types/melanoma-skin-cancer/about/key-statistics.html.

[32] Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin, 76*(5), 378–382. https://doi.org/10.1037/h0031619

[33] Groh, M., Tseng, E., Mahoney, A., & et al. (2023). Evaluating deep neural networks trained on clinical images in dermatology: The DERM dataset and implications for diversity. *The Lancet Digital Health, 5*(3), e158–e168. https://doi.org/10.1016/S2589-7500(22)00284-7.

[34] Winkler, J. K., Fink, C., Toberer, F., & et al. (2019). Association between dermatoscopic application of artificial intelligence for skin cancer recognition and accuracy of dermatologists in a randomized clinical trial. *JAMA Dermatology, 155*(6), 627–634. https://doi.org/10.1001/jamadermatol.2019.1735.

Prediction of Root Canal Treatment Using Machine Learning

Matej Jelenc
Jožef Stefan Institute
Ljubljana, Slovenia
jelenc11matej@gmail.com

Miljana Shulajkovska
Jožef Stefan Institute
Ljubljana, Slovenia
miljana.sulajkovska@ijs.si

Rok Jurič
Odontos, Private Endodontic Practice
Ljubljana, Slovenia
rok.juric@odontos.si

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

Abstract

Root canal treatment is a medical procedure aimed at preventing or treating apical periodontitis, which is an inflammation around the apex of a tooth root. In this study, we analyzed a dataset collected by an experienced practitioner over the course of several years, and developed a forecasting model, based on the XGBoost algorithm, to predict the outcome of the treatment. The trained models achieved a mean area under the receiver-operating-characteristic curve (AUROC) of 0.92 and average precision (AP) of 0.77. We discuss the importance of individual features in view of expert dental knowledge. To assist the practitioner in daily practice, we developed a web-based application to provide an assessment of treatment outcomes.

Keywords

root canal treatment outcome, feature importance, gradient boosting machines

1 Introduction

Apical periodontitis is an inflammation of tissues around the apex of a tooth. It is a major health burden in the general population, with 6% of all teeth showing signs of this condition. Root canal treatment (RCT) is aimed to either prevent the onset of apical periodontitis or to help the tissue to heal if it is already present [13]. Predicting treatment outcomes in RTC is of high interest both to the patients and the dentists, as well as to the insurance companies, as information about the likelihood of successful treatment can lead to better allocation of resources and avoid potentially more invasive procedures, such as tooth removal and its replacement with an implant.

Machine learning has previously been used to study some aspects of the root canal treatment, including association between patient-, tooth- and treatment-level factors and root canal treatment failure [10], predicting root fracture after root canal treatment and crown installation [6], and non-surgical root canal treatment prognosis [2]. In this study, we analyze the data collected by Jurič et al. [13]. This dataset is of special interest since it relies on the RCT patient data obtained by a single experienced practitioner (ensuring a high level of consistency in the treatment approach), as opposed to studies where numerous dentists were treating patients and different choices between them could have resulted in a less representative dataset. The aim of the study was to develop and evaluate an algorithm that predicts the outcome of the RCT, as well as to analyze how robust the algorithm is and which features influence the outcome the most. This study goes hand-in-hand with the study by Jurič et al. [13] where the analysis was conducted solely using statistical methods.

2 Related Work

To our knowledge, utilization of machine learning in endodontics is still relatively unresearched, specifically when predicting treatment outcome only using tabular data. Among the related papers, [10] employs XGBoost to explore the association between patient-, tooth- and treatment-level factors and root canal treatment failure, while [2] used Random Forests (RF), K Nearest Neighbours (KNNs), Logistic Regression (LR) and Naive-Bayes (NB) to predict the outcome of non-surgical root canal treatments, similarly to this study. Paper [8] explores the prediction of treatment longevity using Support Vector Machines (SVMs), LR and NB, while [14] investigates the relation between root canal morphology and root canal treatment using both statistical and machine learning methods, specifically, using RF, SVMs and Gradient Boosting Machines (GBMs). Moreover, papers [19, 18] investigate the prediction of case difficulty and prognosis of endodontic microsurgery, while [6, 9] explore the prediction of root fracture and postoperative pain after root canal treatment. Additionally, multiple papers have been found to investigate root canal treatment outcome or related factors using deep learning (DL) on X-ray images, specifically panoramic or periapical radiographs, such as [3, 22, 11, 1, 5].

3 Data

The dataset analyzed in this study contains treatment details, clinical and radiographic data regarding primary or secondary root canal treatment of mature permanent teeth collected and curated in [13]. Three different types of outcome were determined - clinical, radiographic, and combined, for which both a strict (no clinical or radiographical sign of disease) and loose (only negligible sign of disease) assessment criteria were used. In this paper, only strict assessments were considered and used as prediction targets. All assessments were binary, with 1 representing successfull and 0

Information Society 2025, Ljubljana, Slovenia

https://doi.org/10.70314/is.2025.skui.1849

representing unsuccessfull treatment outcome. The dataset was fairly imbalanced, with 88% of all cases representing successfull radiographic outcome, 92% successfull clinical outcome and 83% successfull combined outcome. The study cohort consisted of 740 patients and 1264 teeth, resulting in 3153 root canal treatment cases and 84 features in total. The majority of features represented either categorical or binary values, such as variables representing gender, tooth type, root canal etc., while variables such as age and working length were treated as continous.

4 Methods

This section outlines the methods used for ranking feature importance and finally training baseline models that can be used as a tool for prediction of root canal treatment outcome.

4.1 Data Preprocessing

First, data regarding second visits was removed, to ensure consistency among cases. Next, features directly dependent or derived from a specific feature were excluded from the dataset to minimize the dimensionality of the data, as well as any post-operative factors that were directly used to determine the treatment outcome. The dataset was further reduced by removing redundant features, which can only have one value or their value is missing for more than 50% of all cases. Similarly, cases for which more than 50% of features are missing were excluded, resulting in 3153 cases and 84 features in total. Lastly, the dataset was preprocessed using label encoding and evenly split into training (80%) and testing (20%) sets. Furthermore, the training set was split into training (80%) and validation (20%) sets when ranking feature importance, to avoid overfitting.

4.2 Model Architecture

For the underlying model, gradient boosting machines were used, specifically the XGBoost algorithm [7], as it remains widely regarded as the state-of-the-art and preferred choice for tabular data tasks, over the more and more popular deep learning algorithms, as shown in [4, 12, 20]. Furthermore, algorithms based on transparent methods, such as decision trees, are strongly preferred for applications in medicine when compared to the "black box" approaches typically associated with deep learning.

4.3 Metrics

Due to the dataset's high imbalance between negative (87%) and positive (13%) cases, standard classification metrics such as accuracy or area under the receiver-operating-characteristic curve (AUROC) can be highly misleading, therefore average precision (AP) was chosen as the key metric for estimating model's performance and ability to produce quality predictions, specifically using the formula:

$$AP = \sum_{i=2}^{n} (R_i - R_{i-1}) \cdot P_i$$

where R_i and P_i are recall and precision at the i-th threshold when testing on n samples [17], while AUROC was only used to provide additional insight when interpreting results.

4.4 Grid Search

To obtain reasonable starting training hyperparameters and a baseline model that utilizes all available information, we performed cross-validated grid-search over a simple manually defined parameter grid, using the scikit-learn library [17].

4.5 Correlation Clustering

When a subset of features in a dataset is highly correlated, standard methods such as feature permutation importance or performing an ablation study often produce inaccurate results, since the model can highly depend only on a specific feature and discard correlated features. Similarly, methods such as SHapley Additive exPlanations (SHAP) [16] or XGBoost's built-in feature importances only account for the contribution of a specific feature to the model's prediction, which can again be misleadingly low due to the feature's correlation to another.

To address this problem, clustering was performed based on the correlation between features. Let $X \in \mathbb{R}^{m \times n}$ represent the dataset with m cases and n features. By calculating the Spearman rank correlation coefficient [15, 17, 23] on X, a symmetric feature correlation matrix $C \in \mathbb{R}^{n \times n}$ was obtained and transformed into a distance matrix $D \in \mathbb{R}^{n \times n}$. To group correlated features, hierarchical clustering using Ward's method [17, 21] was performed on D to obtain a hierarchical clustering tree, which was then flattened into discrete clusters containing features with high absolute correlation.

4.6 Ranking Feature Importance

To determine the significance of a specific feature f, a separate XGBoost model M_f was trained and evaluated on a reduced dataset X_f to obtain baseline results. Next, permutation testing was conducted by permuting the feature f in the testing set and calculating the drop in performance of M_f compared to the baseline results. Each feature was tested 20 times. Lastly, a mean drop and p-value were calculated on the observed performance drops by performing a t-test, where a high mean drop represented high feature importance and a low p-value represented a low chance that the observed drop in performance was caused by an outside factor and not by the random distribution of f in the test set. To ensure that the feature's importance estimation was not corrupted by any correlated features and at the same time account for the feature's possible non-linear connections with other features, while also minimizing the computational cost as much as possible, the reduced dataset X_f was determined as follows.

First, using the model trained on all features (see 4.4), SHAP values [17, 16] were calculated to determine the most contributing feature inside of each cluster. Let $F = \{f_1, \ldots, f_n\}$ represent the set of all features and $S : F \to \mathbb{R}^m$ the transformation that returns SHAP values for a specific feature. The most contributing feature inside of the i-th correlation cluster $C_i = \{f_j \mid j \in I_i\}$ was calculated by taking the feature with the highest mean absolute SHAP value i.e. such $f^* \in C_i$ that $\forall j \in I_i : \overline{|S(f_j)|} \leq \overline{|S(f^*)|}$.

The reduced dataset $X^* \in \mathbb{R}^{m \times r}$, containing only representative features, was then transformed into X_f for a feature $f \in C_i$ by replacing f_i^* by f in X^*. Such approach allows eliminating features highly correlated to f and reduces computational cost by only utilizing the most contributing feature within each cluster, while still accounting for any non-linear connections between f and features in other clusters. The procedure is visualized in Figure 1.

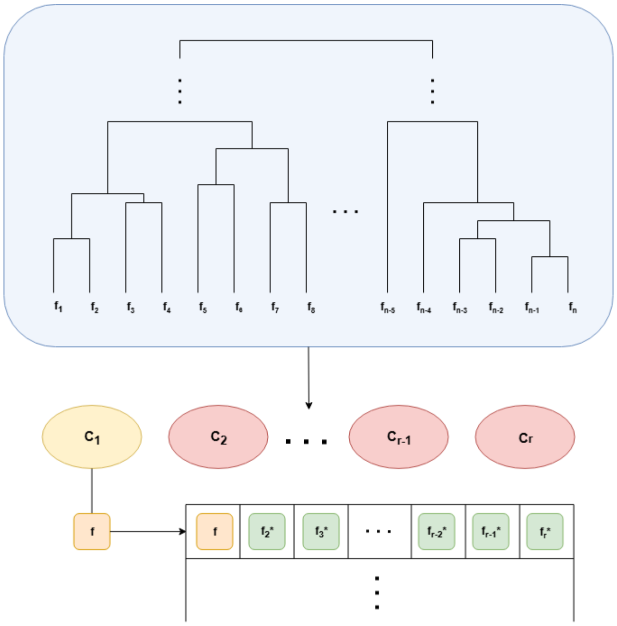


Figure 1: The hierarchical correlation tree is first flattened into clusters $C_1, \ldots, C_r$, for which representative features $f_1^*, \ldots, f_r^*$ define the base dataset X^*, from which we get X_f for $f \in C_i$ by replacing f_i^* by f.

4.7 Evaluation

After obtaining feature importances, features with p-value < 0.05 were deemed as significant. Next, a model using starting parameters found in 4.4 was trained on features belonging in the k-th percentile in terms of feature importance, for k in 1%, 5%, 10%, 25%, 50%, 75%, and 100% (the latter corresponding to all significant features).

5 Results

Figures 2 show the comparison of performances in terms of AP of models trained on different percentiles. The highest performance was achieved when utilizing the entire preprocessed dataset consisting of 84 distinct features in total, achieving AUROC of 0.90 and AP of 0.70 when predicting radiographic outcome, AUROC of 0.94 and AP of 0.86 when predicting clinical outcome and finally AUROC of 0.91 and AP of 0.77 when predicting combined outcome. Out of the 84 chosen features, our method deemed 39 of them significant for radiographic assessment, 54 significant for clinical assessment, and 65 for combined assessment, which produced AUROC of 0.88, 0.85, 0.87 and AP of 0.66, 0.75 and 0.70 respectively.

6 Discussion and Conclusion

Achieving high performance, our paper shows promise in using machine learning techniques for predicting the outcome of endodontic treatments. Moreover, we developed a web application, which allows predicting the outcome of root canal treatments using the models trained on different subsets of data, serving as a tool to assist in assessing the quality and success of a treatment, as well as to give insight for possible further patient care.

Furthermore, all the statistically significant factors found in the original study [13], are found as significant by our method as well. Specifically, "lesion diameter" was found to be the most relevant factor, with "root PAI" and "canal code" being in the top 5%, "tooth type" ("tooth group" and "canal number") in the top 10%, "type of sealer" and "quality of coronal restoration" in the top 25%, "tenderness to periapical palpation" and "quality of root filling" in the top 50% and lastly "injury history" in the top 100% of all significant features. Here, we exclude factors such as "number of visits" and "number of canals per root", since they were not used in this study. Moreover, among the most important factors that this study found and were not accounted for or found as insignificant in [13], are "age" as the second most important factor, "cumulative time" being in the top 5% and "alergic disorders", "working length", "treatment type", "obturation", "PD local", "vertical percussion", "fistulation" and "pain bite" being in the top 25%. Such results suggest that machine learning techniques can perhaps be a better or alternative approach for ranking feature significance in comparison with standard statistical methods such as logistic regression models, since they better account for possible non-linear relationships between different factors and the treatment outcome.

To further refine our approach of selecting significant features, we plan to test different p-values, as the models trained on only significant features achieved a lower performance than the models trained on the entire dataset, with a 5% drop in AUROC and a 7% drop in AP on average, suggesting that there are features which our method deemed insignificant despite enhancing the models' ability to learn and produce accurate results. Future work will also involve analysis of third-party datasets to investigate whether the results obtained in this study are generalizable and to what degree the data collected by a single experienced practitioner is different to a dataset that is typically collected over a course of several years by a number of dentists-in-training. Additionally, we wish to incorporate various explainability techniques, to better justify the models' predictions, in turn giving a deeper insight into how specific factors affect the outcome of root canal treatments as well as better assist a doctor in understanding and interpreting the predicted outcome.

References

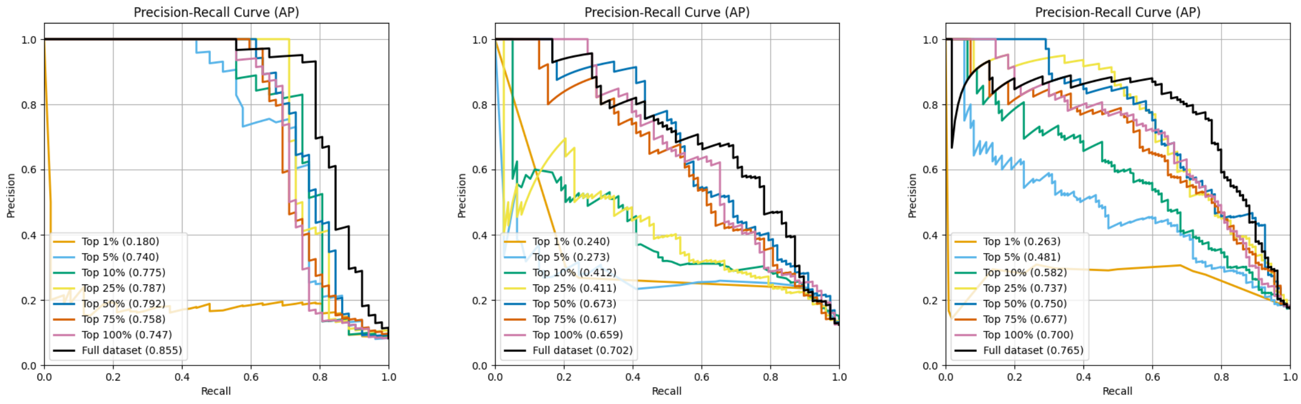
[1] Muhammed Ayhan, İsmail Kayadibi, and Berkehan Aykanat. 2025. Rcfla-yolo: a deep learning-driven framework for the automated assessment of root canal filling quality in periapical radiographs. *BMC Medical Education*, 25, 1, 894. DOI: 10.1186/s12909-025-07483-2.

[2] Catalina Bennasar, Irene García, Yolanda Gonzalez-Cid, Francesc Pérez, and Juan Jiménez. 2023. Second opinion for non-surgical root canal treatment prognosis using machine learning models. *Diagnostics*, 13, 17, 2742. DOI: 10.3390/diagnostics13172742.

[3] Catalina Bennasar, Antonio Nadal-Martínez, Sebastiana Arroyo, Yolanda Gonzalez-Cid, Ángel Arturo López-González, and Pedro Juan Tárraga. 2025. Integrating machine learning and deep learning for predicting non-surgical root canal treatment outcomes using two-dimensional periapical radiographs. *Diagnostics*, 15, 8, 1009. DOI: 10.3390/diagnostics15081009.

[4] Vadim Borisov, Tobias Leemann, Kathrin Seßler, Johannes Haug, Martin Pawelczyk, and Gjergji Kasneci. 2024. Deep neural networks and tabular data: a survey. *IEEE Transactions on Neural Networks and Learning Systems*, 35, 6, 7499–7519. DOI: 10.1109/TNNLS.2022.3229161.

[5] Berrin Çelik, Mehmet Zahid Genç, and Mahmut Emin Çelik. 2025. Evaluation of root canal filling length on periapical radiograph using artificial intelligence. *Oral Radiology*, 41, 1, 102–110. DOI: 10.1007/s11282-024-00781-3.



(a) Strict clinical assessment

(b) Strict radiographic assessment

(c) Strict combined assessment

Figure 2: Average precision (AP) achieved by XGBoost when predicting strict clinical, radiographic and combined assessment, utilizing different subsets of features - all features, all significant features, top 75%, top 50%, top 25%, top 10%, top 5% and top 1% significant features.

[6] Wan-Ting Chang, Hsun-Yu Huang, Tzer-Min Lee, Tsen-Yu Sung, Chun-Hung Yang, and Yung-Ming Kuo. 2024. Predicting root fracture after root canal treatment and crown installation using deep learning. *Journal of Dental Sciences*, 19, 1, 587–593. DOI: 10.1016/j.jds.2023.10.019.

[7] Tianqi Chen and Carlos Guestrin. 2016. Xgboost: a scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (KDD '16). ACM, (Aug. 2016), 785–794. DOI: 10.1145/2939672.2939785.

[8] Pragati Choudhari, Anand Singh Rajawat, and S. B. Goyal. 2023. Longevity recommendation for root canal treatment using machine learning. *Engineering Proceedings*, 59, 1, 193. DOI: 10.3390/engproc2023059193.

[9] Xin Gao, Xing Xin, Zhi Li, and Wei Zhang. 2021. Predicting postoperative pain following root canal treatment by using artificial neural network evaluation. *Scientific Reports*, 11, 1, 17243. DOI: 10.1038/s41598-021-96777-8.

[10] Chantal S. Herbst, Falk Schwendicke, Joachim Krois, and Sascha R. Herbst. 2022. Association between patient-, tooth- and treatment-level factors and root canal treatment failure: a retrospective longitudinal and machine learning study. *Journal of Dentistry*, 117, 103937. DOI: 10.1016/j.jdent.2021.103937.

[11] Sascha Rudolf Herbst, Vinay Pitchika, Joachim Krois, Aleksander Krasowski, and Falk Schwendicke. 2023. Machine learning to predict apical lesions: a cross-sectional and model development study. *Journal of Clinical Medicine*, 12, 17, 5464. DOI: 10.3390/jcm12175464.

[12] Yejin Hwang and Jongwoo Song. 2023. Recent deep learning methods for tabular data. *Communications for Statistical Applications and Methods*, 30, 2, (Mar. 2023), 215–226. DOI: 10.29220/CSAM.2023.30.2.215.

[13] Rok Jurič, G. Vidmar, R. Blagus, and Janja Jan. 2024. Factors associated with the outcome of root canal treatment—a cohort study conducted in a private practice. *International Endodontic Journal*, 57, 4, 377–393. DOI: 10.1111/iej.14022.

[14] Mohmed Isaqali Karobari, Vishnu Priya Veeraraghavan, P. J. Nagarathna, Sudhir Rama Varma, Jayaraj Kodangattil Narayanan, and Santosh R. Patil. 2025. Predictive analysis of root canal morphology in relation to root canal treatment failures: a retrospective study. *Frontiers in Dental Medicine*, 6. DOI: 10.3389/fdmed.2025.1540038.

[15] Maurice G. Kendall and Alan Stuart. 1973. *The Advanced Theory of Statistics, Volume 2: Inference and Relationship.* (1st ed.). See Section 31.18. Charles Griffin, London, UK. ISBN: 978-0852640111.

[16] Scott Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. (2017). https://arxiv.org/abs/1705.07874 arXiv: 1705.07874 [cs.AI].

[17] F. Pedregosa et al. 2011. Scikit-learn: machine learning in python. *Journal of Machine Learning Research*, 12, 2825–2830.

[18] Yang Qu, Zhenzhe Lin, Zhaojing Yang, Haotian Lin, Xiangya Huang, and Lisha Gu. 2022. Machine learning models for prognosis prediction in endodontic microsurgery. *Journal of Dentistry*, 118, 103947. DOI: 10.1016/j.jdent.2022.103947.

[19] Yang Qu, Yiting Wen, Ming Chen, Kailing Guo, Xiangya Huang, and Lisha Gu. 2023. Predicting case difficulty in endodontic microsurgery using machine learning algorithms. *Journal of Dentistry*, 133, 104522. DOI: 10.1016/j.jdent.2023.104522.

[20] Ravid Shwartz-Ziv and Amitai Armon. 2022. Tabular data: deep learning is not all you need. *Information Fusion*, 81, 84–90. DOI: 10.1016/j.inffus.2021.11.011.

[21] Joe H Ward Jr. 1963. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58, 301, 236–244.

[22] Weiwei Wu, Surong Chen, Pan Chen, Min Chen, Yan Yang, Yuan Gao, Jingyu Hu, and Jingzhi Ma. 2024. Identification of root canal morphology in fused-rooted mandibular second molars from x-ray images based on deep learning. *Journal of Endodontics*, 50, 9, 1289–1297.e1. DOI: 10.1016/j.joen.2024.05.014.

[23] Daniel Zwillinger and Stephen Kokoska. 2000. *CRC Standard Probability and Statistics Tables and Formulae.* (1st ed.). Section 14.7. Chapman & Hall/CRC, Boca Raton, FL. ISBN: 978-0-8493-0026-4.

Predictive Maintenance of Machines in LABtop Production Environment

Primož Kocuvan
Department of intelligent systems
"Jožef Stefan" Institute
Ljubljana, Slovenia
primoz.kocuvan@ijs.si

Vinko Longar
Rudolfovo - znanstveno in tehnološko središče
Novo mesto, Slovenia
vinko.longar@rudolfovo.eu

Rok Struna
Rudolfovo - znanstveno in tehnološko središče
Novo mesto, Slovenia
rok.struna@rudolfovo.eu

Abstract

This study investigates predictive maintenance of CNC machinery within the LABtop production environment through the deployment of iCOMOX sensor modules on a compressor and machine spindle. Each module integrates multi-modal sensing capabilities, including vibration, magnetic field, temperature, and acoustic measurements, enabling comprehensive monitoring of machine conditions. Data was collected at five-minute intervals over a 30-day period, resulting in an unlabeled dataset due to the absence of recorded failures or anomalies. The analysis employed unsupervised machine learning techniques, specifically principal component analysis (PCA) for dimensionality reduction and clustering to identify operational patterns. PCA successfully reduced the original 11-dimensional dataset to two principal components, allowing for effective visualization and grouping. The elbow and silhouette methods determined three optimal clusters for both sensors, with one cluster in each case identified as a potential outlier. Results suggest that dense clusters represent normal operation, while outlier clusters may indicate measurement errors or emerging machine faults. Although supervised learning could not yet be applied, future work will integrate fault-labeled data to enable robust predictive maintenance models.

Keywords

predictive maintenance, PCA method, production environment, silhouette analysis, elbow method.

1 Introduction

The increasing complexity of modern production systems demands advanced approaches to machine maintenance in order to minimize downtime, reduce costs, and ensure consistent product quality. Traditional maintenance strategies, such as corrective or preventive maintenance, often fail to provide early warnings of failures and may result in either excessive servicing or unexpected breakdowns. Predictive maintenance, by contrast, leverages sensor data and machine learning techniques to detect patterns, identify anomalies, and forecast potential failures before they occur. This approach not only enhances operational efficiency but also extends the lifetime of critical equipment.,

Within the LABtop production environment (consists of multiple machines in sequence - mostly drilling and cutting machines), predictive maintenance has been explored through the integration of advanced multi-sensor monitoring solutions. For this purpose, the public research institute *Rudolfovo* implemented iCOMOX sensor modules on both the compressor and the spindle of a CNC machine. Each iCOMOX module integrates several sensing elements—vibration, magnetic field, temperature, and acoustic measurements—providing a rich dataset suitable for machine learning–based condition monitoring.

The collected data were acquired over a continuous 30-day period at five-minute intervals. Since no machine failures, temperature anomalies, or bearing defects were recorded during this time, the dataset lacked diagnostic labels and was therefore treated as unlabeled. To address this, unsupervised learning methods were employed to uncover latent structures in the data. Principal component analysis (PCA) was used to reduce the dimensionality of the dataset, while clustering methods were applied to identify patterns and potential anomalies in machine operation. The aim of this study is to evaluate the feasibility of unsupervised learning methods in predictive maintenance for industrial equipment, specifically under conditions where fault-labeled data are unavailable. By analyzing the clustering behavior of sensor signals, this work provides insights into normal operating regimes and potential deviations that may correspond to early indicators of faults or measurement errors. Future work will incorporate supervised learning techniques once labeled fault data become available, enabling the development of robust predictive models.

http://doi.org/10.70314/is.2025.skui.0545

2 Related Work

The field of predictive maintenance (PdM) has advanced considerably, with strong emphasis on unsupervised learning methods for anomaly detection and health assessment when labeled failure data are unavailable. PdM has been shown to significantly reduce maintenance costs, decrease unexpected downtime, and enhance equipment reliability [1]. Multi-sensor monitoring platforms such as iCOMOX have emerged as versatile tools for industrial condition monitoring. These devices integrate vibration, magnetic field, temperature, and acoustic sensors into a compact, industrial-grade package capable of edge analytics and cloud integration [2–5].

Such systems enable continuous monitoring of machine health and support the implementation of predictive maintenance strategies in Industry 4.0 environments. From a methodological perspective, unsupervised learning techniques, such as principal component analysis (PCA) and clustering, are widely applied for exploratory data analysis, dimensionality reduction, and anomaly detection. A comprehensive survey highlights the breadth and maturity of these techniques across domains [6]. Clustering methods including k-means, DBSCAN, and OPTICS are instrumental in grouping operational states and unveiling deviations that may signify incipient failures [7].

Hybrid methods combining PCA with clustering have proven effective in enhancing fault detection capabilities. For example, a railcar health monitoring system employing DBSCAN clustering with PCA achieved fault detection accuracy of 96.4% [8]. Similarly, kernel PCA has been applied to construct health indices for unsupervised prognostics [9]. In compressor maintenance, incorporating clustering-derived features into supervised classifiers improved predictive accuracy by 4.9% and reduced training time by 23% [10]. Several studies also propose frameworks that integrate unsupervised learning with IoT and Big Data infrastructures, enabling scalable predictive maintenance solutions across industrial environments [11]. These works demonstrate the feasibility of extracting actionable health indicators from unlabeled sensor data and underscore the critical role of advanced analytics in industrial condition monitoring.

3 Methodology

3.1 Data Acquisition

Two iCOMOX sensor modules were installed on critical machine components within the LABtop production system: the spindle of a CNC machine and the air compressor. Each sensor module integrates vibration, magnetic field, temperature, and acoustic sensing elements, thereby providing multimodal monitoring capabilities. Data were sampled at 5-minute intervals over a continuous 30-day observation period, resulting in an unlabeled dataset due to the absence of recorded failures, anomalies, or maintenance events.

3.2 Data Preprocessing

Raw signals from the iCOMOX modules were aggregated into feature vectors, yielding an 11-dimensional dataset. Standard preprocessing steps included: normalization of features to remove scaling effects, filtering to reduce noise (particularly in the acoustic and vibration signals), and synchronization of multimodal sensor streams.

3.3 Dimensionality Reduction

To facilitate visualization and clustering, dimensionality reduction was performed. Multiple techniques (e.g., t-SNE, Isomap, and autoencoders) were evaluated; however, Principal Component Analysis (PCA) demonstrated superior stability and interpretability. The data were reduced from 11 to 2 principal components, which captured the majority of the variance and allowed effective 2D representation.

3.4 Clustering Analysis

Clustering was applied to the reduced dataset to uncover hidden structures and potential anomalies. The elbow method (*Figure 1*) and silhouette coefficient (*Figure 2*) were employed to determine the optimal number of clusters. Based on these metrics, three clusters were identified for each sensor dataset.

The analysis was conducted separately for the two sensor modules (iCOMOX1 on the spindle and iCOMOX2 on the compressor). Outlier clusters were identified and highlighted for subsequent interpretation.

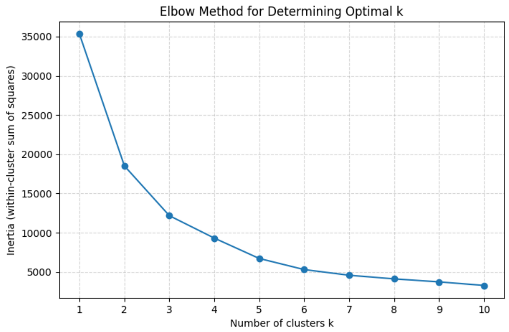


Figure 1: Elbow method

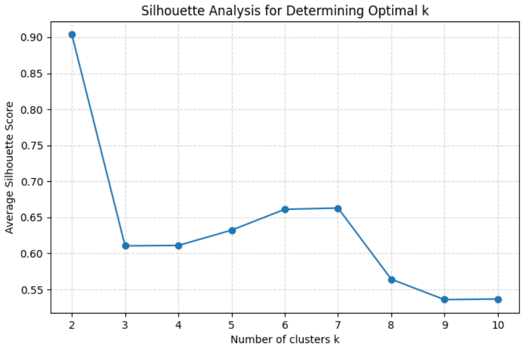


Figure 2: Silhouette coefficient

4 Results

PCA successfully compressed the 11-dimensional dataset into a two-dimensional space. The first two principal components explained the majority of the variance (>80%), enabling effective visualization of patterns in machine behavior. *Figure 3* illustrates the scatter plot for iCOMOX1. Three distinct clusters are visible, with Cluster 1 (highlighted in orange) showing divergence from the main operating regime. *Figure 4* presents the scatter plot for iCOMOX2, where Cluster 2 (highlighted in green) emerges as an outlier relative to the normal operating clusters.

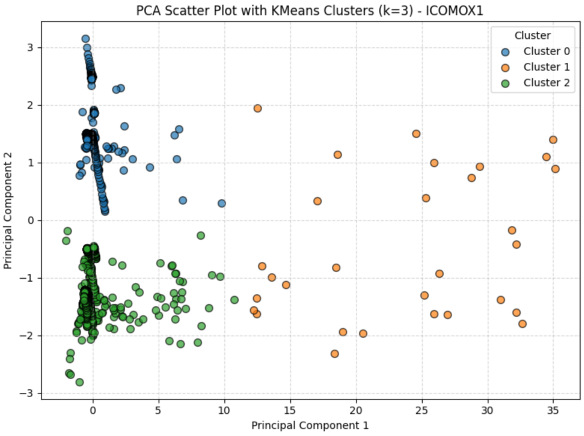


Figure 3: PCA Scatter plot for ICOMOX1

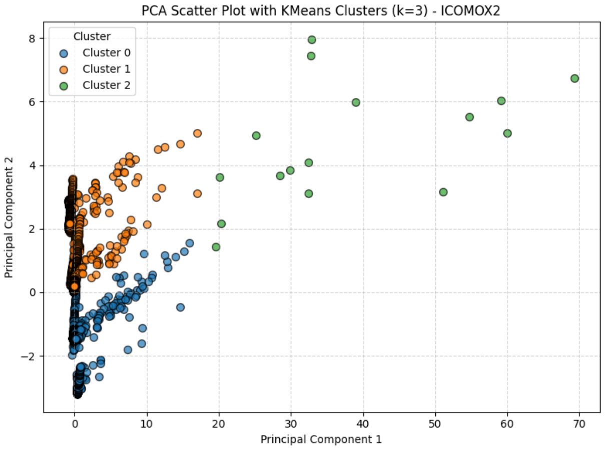


Figure 4: PCA Scatter plot for ICOMOX2

Clusters containing densely grouped points correspond to normal operating conditions of the CNC spindle and compressor. The outlier clusters, however, represent either:

- sensor noise or measurement anomalies (e.g., transient vibration spikes or acoustic distortions), or
- incipient machine faults, which could not be conclusively confirmed due to the absence of ground-truth failure data.

- PCA combined with clustering effectively distinguished between normal operation and anomalous behavior.
- Both sensor datasets (iCOMOX1 and iCOMOX2) revealed three clusters, with one consistently standing out as an outlier.
- Without diagnostic labels, these outliers cannot be definitively classified as machine faults, but their presence highlights potential events of interest for further investigation.
- The results validate the feasibility of unsupervised learning for predictive maintenance in environments lacking labeled fault data.

5 Discussion

The findings from this study demonstrate the viability of unsupervised learning methods in particular PCA and clustering for analyzing unlabeled condition-monitoring data in industrial environments. By reducing an 11-dimensional dataset to two principal components, it was possible to visualize operational states and uncover outlier clusters that may correspond to anomalous machine behavior. This outcome aligns with previous work emphasizing the effectiveness of dimensionality reduction and clustering in predictive maintenance tasks where labeled fault data are limited or unavailable [6,8,9].

The observation of three clusters for both the spindle (iCOMOX1) and compressor (iCOMOX2) highlights the presence of distinct operating regimes within the LABtop system. The fact that one cluster consistently emerged as an outlier suggests potential precursors to faults or, alternatively, sensor-related anomalies. While conclusive interpretation requires diagnostic labels, the clustering nevertheless provides an essential first step toward identifying patterns that can later inform supervised learning models once fault data become available.

Compared to related studies, the present results confirm trends reported in railcar health monitoring [8] and compressor maintenance [10], where unsupervised approaches successfully revealed structural patterns in the absence of labeled datasets. The advantage of PCA lies in its ability to preserve variance while simplifying visualization, which proved more effective than alternative reduction methods considered here (e.g., t-SNE or Isomap). This echoes findings from other industrial applications where PCA has served as a reliable baseline for anomaly detection [9].

An important implication is that multi-sensor platforms such as iCOMOX provide the richness of data required for advanced analytics. The combination of vibration, acoustic, magnetic field, and temperature measurements enables detection of subtle variations that might not be visible through single-sensor monitoring. As highlighted in prior work [2–5], the integration of multimodal data streams significantly strengthens predictive maintenance frameworks by improving robustness and interpretability.

Nevertheless, this study also underscores the limitations of unsupervised learning. Without failure labels, it is not possible to conclusively distinguish between anomalies arising from true machine faults and those caused by sensor noise or environmental conditions. This limitation has been widely noted in the literature [6,11]. Future work should therefore focus on generating labeled datasets through controlled fault injection or long-term monitoring until natural failures occur. Such datasets would enable supervised and hybrid learning approaches, which have shown promise in achieving higher predictive accuracy and more actionable decision support [1,10].

In summary, the present analysis validates the potential of unsupervised learning for predictive maintenance in data-scarce environments. While preliminary, the results establish a methodological foundation for extending condition monitoring at LABtop to more advanced machine learning pipelines, ultimately contributing to early fault detection, reduced downtime, and optimized maintenance planning.

6 Future Work

The present study establishes a foundation for predictive maintenance at LABtop using unsupervised learning methods; however, several directions remain open for further investigation.

First, the absence of diagnostic labels limited this study to exploratory clustering and anomaly detection. Future work will prioritize the collection of labeled datasets through either (i) controlled fault injection experiments on non-critical test equipment or (ii) extended operational monitoring until natural failures occur. The availability of labeled fault data will enable the application of supervised learning and hybrid approaches, combining clustering-derived features with classification models to improve fault detection accuracy and reliability, as demonstrated in recent compressor studies [10]. Second, while PCA provided an effective means of dimensionality reduction, more advanced techniques such as kernel PCA, autoencoders, and variational autoencoders should be investigated. These methods may capture nonlinear relationships in the sensor data that PCA cannot, potentially yielding richer health indicators and more precise separation of operational regimes [9]. Third, the present work focused primarily on offline analysis. Future research should extend to real-time streaming analytics, leveraging the edge-processing capabilities of the iCOMOX platform [2–5]. Deploying online anomaly detection models would allow immediate identification of abnormal conditions and facilitate proactive maintenance decisions.

Fourth, integration with IoT and cloud-based platforms remains a key step toward scalable deployment. By embedding unsupervised learning models into Industry 4.0 architectures, LABtop can benefit from centralized monitoring, cross-machine comparisons, and fleet-level anomaly detection, as highlighted in existing frameworks [11].

Finally, interpretability remains an essential concern. Future efforts will explore explainable AI (XAI) techniques to provide actionable insights into why certain clusters or anomalies are flagged, thereby enhancing operator trust and enabling domain experts to validate and refine the models.

Acknowledgments

The research was supported by DIGITOP project which is funded by Ministry of Higher Education, Science and Innovation of Slovenia, Slovenian Research and Innovation Agency, and EU NextGenerationEU under Grant TN-06-0106. We thank prof. dr. Matjaž Gams for proof-reading the article and mentorship support within DIGITOP project.

References

[1] Abdeldjalil Benhanifia, Zied Ben Cheikh, Paulo Moura Oliveira, Antonio Valente, José Lima. *Systematic review of predictive maintenance practices in the manufacturing sector*. Intelligent Systems with Applications, Volume 26, 2025, Article 200501. ISSN 2667-3053. https://doi.org/10.1016/j.iswa.2025.200501

[2] RS Components, *iCOMOX Intelligent Condition Monitoring Box – Product Datasheet*, 2019. Available: https://docs.rs-online.com/c878/A700000007538369.pdf, Accessed 25.8.2025

[3] EE Times Europe, *Arrow introduces new Shiratech iCOMOX condition-based monitoring products*, 2019. Available: https://www.eetimes.eu/press-releases/arrow-introduces-new-shiratech-icomox-condition-based-monitoring-products/, Accessed 25.8.2025

[4] EBOM, *New Shiratech iCOMOX sensor-to-cloud platform cuts time-to-market for intelligent condition monitoring*, 2019. Available: https://www.ebom.com/new-shiratech-icomox-sensor-to-cloud-platform-cuts-time-to-market-for-intelligent-condition-monitoring/, Accessed 25.8.2025

[5] Sensor+Test, *iCOMOX – Condition Monitoring Box*, 2023. Available:https://www.sensor-test.de/assets/Fairs/2023/ProductNews/PDFs/iCOMOX.pdf, Accessed 25.8.2025

[6] K. Taha, "Semi-supervised and un-supervised clustering: A review and experimental evaluation," *Information Systems*, vol. 114, p. 102178, 2023. doi: 10.1016/j.is.2023.102178

[7] GopenAI Blog, *Predictive maintenance with unsupervised machine learning algorithms*, 2020. Available: (Blog.gopenai.com), Accessed 25.8.2025

[8] M. Ejlali, E. Arian, S. Taghiyeh, K. Chambers, A. H. Sadeghi, D. Cakdi, and R. B. Handfield, "Developing Hybrid Machine Learning Models to Assign Health Score to Railcar Fleets for Optimal Decision Making," *arXiv preprint* arXiv:2301.08877, 2023.

[9] Z. Chen et al., "Health Index Construction Based on Kernel PCA for Equipment Prognostics," *Control Engineering Practice*, vol. 126, 2022.

[10] A. Salazar et al., "Unsupervised Feature Extraction for Compressor Predictive Maintenance Using Clustering and Supervised Learning," *arXiv*, 2024.

[11] Nota, Giancarlo, Nota, Francesco, Toro Lazo, Alonso Nastasia, Michele. (2024). A framework for unsupervised learning and predictive maintenance in Industry 4.0. International Journal of Industrial Engineering and Management. 15. 304-319. 10.24867/IJIEM-2024-4-365.

Machine Learning for Cutting Tool Wear Detection: A Multi-Dataset Benchmark Study Toward Predictive Maintenance

Žiga Kolar
ziga.kolar@ijs.si
Jožef Stefan Institute
Ljubljana, Slovenia

Thibault Comte
thibault.comte@universite-paris-saclay.fr
Universite Paris-Saclay
Paris, France

Yanny Hassani
yanny.hassani@universite-paris-saclay.fr
Universite Paris-Saclay
Paris, France

Hugues Louvancour
hugues.louv@gmail.com
Universite Paris-Saclay
Paris, France

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

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

Abstract

This student paper investigates the use of machine learning techniques to automate the detection of tool wear in cutting machines, replacing manual monitoring with intelligent, data-driven solutions. Although the proposed ML methods are standard in predictive maintenance, our contribution lies in providing the systematic multi-dataset benchmark tailored for direct transfer to industrial environments. This establishes a reproducible baseline before deploying and validating on real UNIOR data. As part of the project, and in anticipation of collecting real-world accelerometer data from industrial machines, we conducted a series of benchmarking experiments using five publicly available datasets that include accelerometer and audio signals under various wear-related conditions. The datasets cover a variety of industrial contexts and labeling schemes, allowing us to assess different preprocessing strategies and classification models such as Random Forests, 1D Convolutional Neural Networks, and Long Short-Term Memory networks. Our best results—an F1-score of 0.9949—were achieved using an LSTM model on a vibration dataset simulating fault conditions. These findings highlight the strong potential of AI for predictive maintenance and lay the groundwork for transferring the developed pipelines to the system once real data become available. Future work will focus on real-time wear detection and model deployment within live production environments.

Keywords

accelerometer, neural networks, machine learning, cutting tool

1 Introduction

This student paper presents the work carried out by Thibault Comte, Hugues Louvancour, and Yanny Hassani on the UNIOR project, under the mentorship of Žiga Kolar, prof. dr. Matjaž Gams for Jozef Stefan Institute, and Joze Ravnican for Unior. The objective of the UNIOR project is to detect when a cutting machine becomes worn out by analyzing sensor signals, specifically accelerometer data along the x, y, and z axes. An accelerometer is mounted on the cutting machine to monitor vibrations occurring during the cutting process. Currently, the detection of wear is performed manually by a human operator. By leveraging artificial intelligence (AI) and machine learning (ML), this process can be automated, making it both easier and more efficient.

While awaiting the company to complete the necessary paperwork and acquire and install the accelerometer on the cutting machine, we identified similar publicly available datasets and conducted several machine learning experiments using them.

2 Related Work

This section briefly surveys recent research on the use of artificial intelligence (AI) techniques for tool wear monitoring in manufacturing processes such as milling, turning, and drilling. Munaro et al. [2] provide a systematic review of 77 studies, contrasting offline and online monitoring methods. Online approaches leveraging sensor data—such as force, vibration, acoustic emission, and power—are enhanced by AI models like SVMs, ANNs, CNNs, and LSTMs, offering accuracies above 90% and industrial relevance. Sieberg et al. [5] demonstrate CNN-based classification of wear mechanisms from SEM images, achieving 73% test accuracy. They emphasize dataset balance and magnification consistency as critical challenges. Colantonio et al. Shah et al. [4] argue for ML's superiority over physics-based models in wear prediction, underscoring ANN's predictive strength when supplied with high-quality data and standardized evaluation methods. Recent studies also explore multimodal sensor fusion, combining accelerometer, acoustic, and force signals to improve robustness [8]. Specifically, transfer learning has been shown effective for adapting models trained on laboratory data to industrial machines [8].

Unlike previous reviews such as Munaro et al. [2], which survey the field, our work provides a systematic multi-dataset experimental comparison across three different sensor modalities (accelerometer, vibration, audio) using standardized pipelines. This benchmarking is not only descriptive but forms the basis for industrial transfer to UNIOR's production line, bridging academic datasets with real machine applications.

3 Datasets

This section describes five different datasets that were identified—four containing accelerometer data and one featuring audio recordings.

Information Society 2025, Ljubljana, Slovenia

https://doi.org/10.70314/is.2025.skui.6224

3.1 Bosch CNC Machining Dataset

The Bosch CNC Machining dataset consists of real-world industrial vibration data collected from a brownfield CNC milling machine. Acceleration was measured using a tri-axial Bosch CISS sensor mounted inside the machine, recording the X, Y, and Z axes at a sampling rate of 2 kHz. Both normal and anomalous data were collected across six distinct timeframes, each spanning six months between October 2018 and August 2021, with appropriate labeling. Data were collected from three distinct CNC milling machines, each executing 15 processes [7]. A total of 1,702 samples were obtained, with each labeled as either "good" or "bad." The distribution of labels was 95.9% good and 4.1% bad.

3.2 Cutting Tool Wear Audio Dataset

This dataset comprises 1,488 ten-second .wav audio recordings of cutting tool wear collected at two spindle speeds: 520 RPM and 635 RPM. Each audio recording is labeled as either “BASE” (machine running without cutting), “FRESH” (sharp cutting tool), “MODERATE” (moderately worn tool), or “BROKEN” (broken or fully worn tool). The “FRESH,” “MODERATE,” and “BROKEN” labels were specifically chosen to simulate real cutting conditions, focusing on scenarios where the machine is actively engaged in material removal. In total, the dataset includes 400 “FRESH” samples, 376 “MODERATE” samples, and 362 “BROKEN” samples across both spindle speeds, offering a nearly balanced distribution well-suited for ML applications. Audio records had different lengths. No artificial background noise was added to the recordings. All cutting tools used were 16 mm end-mill cutters, and the workpiece material was mild steel [6].

3.3 Turning Dataset for Chatter

This dataset contains sensor signals collected from multiple cutting tests using a range of measurement devices, including two perpendicular single-axis accelerometers, a tri-axial accelerometer, a microphone, and a laser tachometer. Both raw sensor data and processed, labeled data from one channel of the tri-axial accelerometer are provided. There were four labels used: no-chatter, intermediate chatter, chatter, and unknown. The dataset contains a total of 117 signals, with the following label distribution: 51 labeled as no-chatter, 19 as intermediate chatter, 22 as chatter, and 25 as unknown. Data were collected under four distinct cutting configurations, defined by varying the stick out distance—the distance from the heel of the boring rod to the back face of the tool holder. The four stickout distances used were 5.08 cm (2 inches), 6.35 cm (2.5 inches), 8.89 cm (3.5 inches), and 11.43 cm (4.5 inches) [8].

3.4 UCI Accelerometer Dataset

To simulate motor vibrations, a 12 cm Akasa AK-FN059 Viper cooling fan was modified by attaching weights to its blades, and an MMA8452Q accelerometer was mounted to capture vibration data. An artificial neural network was then used to predict motor failure time based on this data. Three distinct vibration scenarios were generated by varying the placement of two weight pieces on the fan blades: (1) Red – normal configuration, with weights on neighboring blades; (2) Blue – perpendicular configuration, with weights on blades 90° apart; and (3) Green – opposite configuration, with weights on opposite blades. For each of the three weight configurations, vibration data was collected every 20 ms over a 1-minute interval per speed, resulting in 3,000 records per speed. In total, the dataset contains 153,000 vibration records from the simulation model [3].

3.5 Vibrations Dataset

This dataset contains vibrational data collected to support early fault diagnosis in machinery The data was gathered using an SG-Link tri-axial accelerometer sensor (by MICROSTRAIN Corporation) at a sampling rate of 679 samples per second for each of the three axes: axial (z), horizontal (x), and vertical (y). Experiments were conducted in the Mechanical Vibration Laboratory at the Mechanical Engineering Department of the University of Engineering and Technology (UET), Taxila. The setup simulated four distinct machine conditions: normal, cracking, offset pulley, and wear states, using a test rig designed for fault simulation [1].

4 Methodology and Results

This section outlines the methodology used for each dataset, focusing on multiclass classification. Various preprocessing techniques and machine learning algorithms were applied.

4.1 Bosch CNC Machining Dataset

The Bosch CNC Machining Dataset contains 95.9% good signals and 4.1% bad signals. The objective was to develop a binary classification model that outperforms a naive baseline, which achieves 95.9% accuracy simply by always predicting a signal as good.

Two approaches were tested on the Bosch CNC Machining dataset. The first approach applied random undersampling, which balances class distribution by randomly removing samples from the majority class while leaving the minority class unchanged. Since the majority class accounted for 95.9% of the data, this step was essential to prevent the model from defaulting to majority-class predictions. After applying the random undersampling, the Random forest model was used for binary classification. This method achieved 99% accuracy on 5-fold cross validation, providing a 3.1% improvement over the naive baseline model.

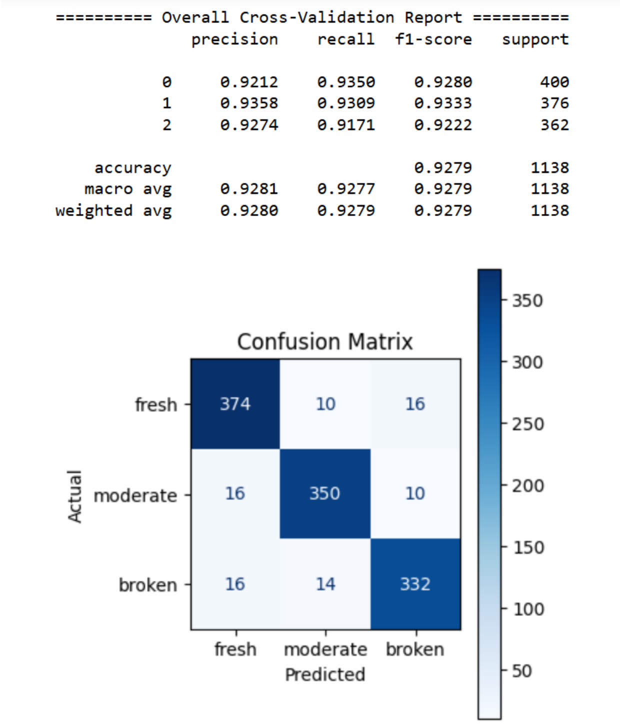
Different preprocessing strategies were necessary due to differences in data formats, sampling rates, and class balance across datasets. For example, in the Bosch dataset, random undersampling was applied only on the training folds during 5-fold CV to avoid information leakage.

In the second approach, features were initially extracted using two 1D Convolutional layers followed by two Max Pooling layers. To augment the data, random Gaussian noise was added to the signals, effectively doubling the size of the training set. A binary classification model using Random Forest was then trained on this augmented dataset. This model achieved a high accuracy of 0.996 under 5-fold cross-validation, outperforming the naive baseline by 3.7%.

McNemar’s test was applied between competing models on each dataset. Significant differences ($p < 0.05$) were observed between CNN and Random Forest on the Bosch dataset, confirming that improvements are not due to random variation.

4.2 Cutting Tool Wear Audio Dataset

The Cutting Tool Wear Audio Dataset contained 400 “FRESH”, 376 “MODERATE”, and 362 “BROKEN” samples across two spindle speeds, requiring a multi-class classification approach. Since the signals varied in length, we first identified the longest signal (48000 samples) and zero-padded shorter signals to match this length. To improve model accuracy, this maximum length was



```
========== Overall Cross-Validation Report ==========
              precision    recall  f1-score   support

           0     0.9212    0.9350    0.9280       400
           1     0.9358    0.9309    0.9333       376
           2     0.9274    0.9171    0.9222       362

    accuracy                         0.9279      1138
   macro avg     0.9281    0.9277    0.9279      1138
weighted avg     0.9280    0.9279    0.9279      1138
```

Confusion Matrix

Actual \ Predicted	fresh	moderate	broken
fresh	374	10	16
moderate	16	350	10
broken	16	14	332

Figure 1: 5-Fold cross validation report and confusion matrix for Cutting Tool Wear Audio dataset.

later reduced. The model architecture included two 1D Convolutional layers and two 1D Max Pooling layers to reduce the dimensionality of the data while preserving essential features.

The output from the upper layers served as input to a feature selection algorithm, which identified the 96 most relevant features out of a total of 2048. These selected features were then used by a Random Forest classifier to predict the final label.

The best model for this dataset achieved 0.9279 (+/- 0.01) accuracy and 0.9279 F1 score on 5-fold cross validation. Results (precision, recall, F1-score and accuracy) are presented on Figure 1.

4.3 Turning Dataset for Chatter

Since each signal varies in length and can be quite long, an approach based on extracting time-domain and frequency-domain features was implemented. This method preserves essential information from the original signals while significantly reducing dimensionality, making the data more suitable for ML algorithms.

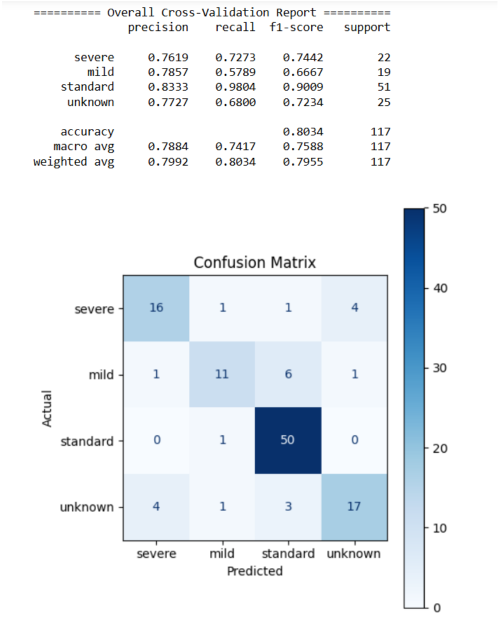
The following approach combines signal segmentation and frequency-domain feature extraction to summarize the spectral characteristics of a time-series signal. First, it divides the input signal into overlapping or non-overlapping fixed-size windows using a sliding window technique, where each segment is of 10000 windows length and the shift between consecutive segments is determined by step size, which in this case is 5000. This allows for localized analysis of signal dynamics over time.

Next, we applied the Fast Fourier Transform (FFT) to each segment, converting the time-domain signal into its frequency-domain representation. It computes the magnitude spectrum for each segment and then averages the spectral magnitudes across all segments to obtain a single, representative frequency-domain feature vector. This results in a compact yet informative summary that captures the dominant frequency components of the entire signal, while accounting for temporal variation through segmentation.

Furthermore, 11 additional features were extracted from the raw signal, including the mean, standard deviation, minimum, maximum, and median of the frequency values. These features capture the signal's central tendency and variability, providing a statistical summary of its frequency content. The 25th and 75th percentiles further quantify the signal's interquartile range, highlighting its variability and robustness to outliers. Root mean square (RMS) provides a measure of the signal's overall power. Skewness and kurtosis describe the asymmetry and peakedness of the distribution, respectively, offering insights into the signal's shape beyond basic statistics. Finally, zero crossings count the number of times the signal crosses the zero axis, serving as an indicator of frequency content and signal complexity. Together, these features form a rich representation for classification tasks involving time-frequency signals.

In total, there were 268 features (257 FFT features and 11 additional features) and 117 samples. A feature selection technique was applied to further reduce the number of features. 140 best features were selected and used as input for Random Forest classifier.

The best model for this dataset achieved 0.80 (+/-0.06) accuracy and 0.7588 F1-score on 5-fold cross validation. Results (precision, recall, F1-score and accuracy) are depicted on Figure 2.



```
========== Overall Cross-Validation Report ==========
              precision    recall  f1-score   support

      severe     0.7619    0.7273    0.7442        22
        mild     0.7857    0.5789    0.6667        19
    standard     0.8333    0.9804    0.9009        51
     unknown     0.7727    0.6800    0.7234        25

    accuracy                         0.8034       117
   macro avg     0.7884    0.7417    0.7588       117
weighted avg     0.7992    0.8034    0.7955       117
```

Confusion Matrix

Actual \ Predicted	severe	mild	standard	unknown
severe	16	1	1	4
mild	1	11	6	1
standard	0	1	50	0
unknown	4	1	3	17

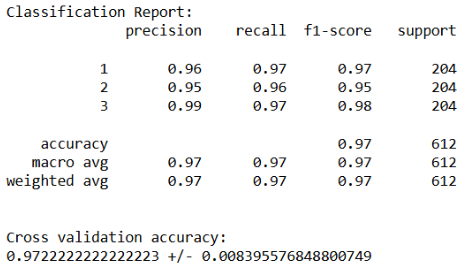
Figure 2: 5-Fold cross validation report and confusion matrix for Turning dataset for Chatter.

4.4 UCI Accelerometer Dataset

This method implements a complete machine learning pipeline for classifying time-series accelerometer data using features extracted from both the time and frequency domains. Data is first loaded from a CSV file, where each row contains an activity label and raw X, Y, and Z accelerometer readings. The signal is segmented into non-overlapping windows of fixed size (50 samples, corresponding to 1 second at 50 Hz), and only windows with consistent activity labels are retained for supervised learning.

Next, time-domain and frequency-domain features were extracted from a signal. Time-domain features include basic statistics (mean, standard deviation, min, max, median), RMS, peak-to-peak range, skewness, kurtosis, zero-crossing rate, signal energy, and crest factor. Frequency-domain features are extracted via FFT and include spectral centroid, spectral spread, peak frequency, and energy in predefined low (0–5 Hz) and high (10–25 Hz) frequency bands.

This feature-rich representation is passed through a machine learning pipeline that includes feature scaling, univariate feature selection (Select K Best ANOVA F-statistical method), and classification using a Random Forest classifier. The best model for this dataset achieved 0.972 (+/-0.008) accuracy and 0.97 F1-score on 5-fold cross validation. Results (precision, recall, F1-score and accuracy) are depicted on Figure 3.



```
Classification Report:
              precision    recall  f1-score   support

           1       0.96      0.97      0.97       204
           2       0.95      0.96      0.95       204
           3       0.99      0.97      0.98       204

    accuracy                           0.97       612
   macro avg       0.97      0.97      0.97       612
weighted avg       0.97      0.97      0.97       612


Cross validation accuracy:
0.9722222222222223 +/- 0.008395576848800749
```

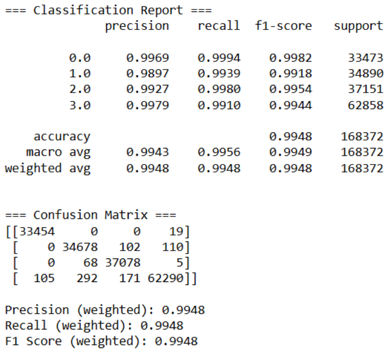
Figure 3: 5-Fold cross validation report and confusion matrix for UCI Accelerometer dataset.

4.5 Vibrations Dataset

In this method the time series data was effectively segmented into overlapping windows of fixed length 226. A total of 168,372 samples were generated, providing a sufficient amount of data for training deep learning models. A Long Short-Term Memory (LSTM) neural network was chosen due to its effectiveness in handling sequential data. The network architecture consisted of two LSTM layers with 128 and 64 units, respectively, along with two Dropout layers incorporated to reduce the risk of overfitting and improve generalization. This method achieved the best performance to date, reaching an accuracy of 0.9948 (+/-0.005) in 5-fold cross-validation and an F1-score of 0.9949. The results are presented on Figure 4.

5 Conclusion

This student paper explored machine learning for automated cutting tool wear detection. Using five public datasets and models such as Random Forests, CNNs, and LSTMs, we achieved strong performance, notably 0.9949 F1 on the Vibrations dataset. These benchmarks highlight ML's potential for predictive maintenance and provide ready-to-deploy pipelines for future industrial data. Future work will focus on validating the model on industrial machines, optimizing its performance, and deploying it in real-time. Additionally, for ordered domains like the Cutting Tool Wear Audio dataset, misclassifications should not be penalized equally (e.g., "FRESH" -> "MODERATE" vs. "FRESH" -> "BROKEN"). Thus, future research will explore ordinal metrics, such as weighted accuracy or quadratic weighted kappa.



```
=== Classification Report ===
              precision    recall  f1-score   support

         0.0     0.9969    0.9994    0.9982     33473
         1.0     0.9897    0.9939    0.9918     34890
         2.0     0.9927    0.9980    0.9954     37151
         3.0     0.9979    0.9910    0.9944     62858

    accuracy                         0.9948    168372
   macro avg     0.9943    0.9956    0.9949    168372
weighted avg     0.9948    0.9948    0.9948    168372


=== Confusion Matrix ===
[[33454     0     0    19]
 [    0 34678   102   110]
 [    0    68 37078     5]
 [  105   292   171 62290]]

Precision (weighted): 0.9948
Recall (weighted): 0.9948
F1 Score (weighted): 0.9948
```

Figure 4: 5-Fold cross validation report and confusion matrix for Vibration dataset.

This study has several limitations. First, the datasets used are publicly available and may not fully capture the variability of industrial machining environments. Second, in some cases class balance was artificially enforced via undersampling, which could affect generalizability. Third, we recognize that the lack of direct industrial validation is a current limitation. However, our pipelines were designed for immediate deployment once the company's accelerometers are installed, ensuring direct continuity from these benchmark studies to industrial application. This study therefore serves as a reproducible foundation rather than a final industrial deployment. Partial validation experiments with UNIOR's machines are planned as the next project stage.

Acknowledgements

The authors acknowledge funding support from the company UNIOR for the GREMO LIGHTWEIGHT project. The authors also acknowledge the funding from the Slovenian Research and Innovation Agency (ARIS), Grant (PR-10495) and Basic core funding P2-0209.

References

[1] Muhammad Umar Khan, Muhammad Atif Imtiaz, Sumair Aziz, Zeeshan Kareem, Athar Waseem, and Muhammad Ammar Akram. 2019. System design for early fault diagnosis of machines using vibration features. In *2019 International Conference on Power Generation Systems and Renewable Energy Technologies (PGSRET)*. IEEE, 1–6.

[2] Roberto Munaro, Aldo Attanasio, and Antonio Del Prete. 2023. Tool wear monitoring with artificial intelligence methods: a review. *Journal of Manufacturing and Materials Processing*, 7, 4, 129. DOI: 10.3390/jmmp7040129.

[3] Gustavo Scalabrini Sampaio, Arnaldo Rabello de Aguiar Vallim Filho, Leilton Santos da Silva, and Leandro Augusto da Silva. 2019. Prediction of motor failure time using an artificial neural network. *Sensors*, 19, 19, 4342.

[4] Raj Shah, Nikhil Pai, Gavin Thomas, Swarn Jha, Vikram Mittal, Khosro Shirvni, and Hong Liang. 2024. Machine learning in wear prediction. *Journal of Tribology*, 147, 4, (Nov. 2024), 040801. eprint: https://asmedigitalcollection.asme.org/tribology/article-pdf/147/4/040801/7400649/trib_147_4_040801.pdf. DOI: 10.1115/1.4066865.

[5] Philipp Maximilian Sieberg, Dzhem Kurtulan, and Stefanie Hanke. 2022. Wear mechanism classification using artificial intelligence. *Materials*, 15, 7, 2358. DOI: 10.3390/ma15072358.

[6] Nachiket Soni, Amit Kumar, and Hardik Patel. 2023. Acoustic analysis of cutting tool vibrations of machines for anomaly detection and predictive maintenance. In *2023 IEEE 11th Region 10 Humanitarian Technology Conference (R10-HTC)*. IEEE, 43–46.

[7] Mohamed-Ali Tnani, Michael Feil, and Klaus Diepold. 2022. Smart data collection system for brownfield cnc milling machines: a new benchmark dataset for data-driven machine monitoring. *Procedia CIRP*, 107, 131–136.

[8] Melih C Yesilli, Firas A Khasawneh, and Andreas Otto. 2020. On transfer learning for chatter detection in turning using wavelet packet transform and ensemble empirical mode decomposition. *CIRP Journal of Manufacturing Science and Technology*, 28, 118–135.

Extracting Structured Information About Food Loss and Waste Measurement Practices Using Large Language Models: A Feasibility Study

Junoš Lukan
junos.lukan@ijs.si
Jožef Stefan Institute
Department of Intelligent Systems
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia

Maori Inagawa
maoriinagawa@keio.jp
Jožef Stefan Institute
Department of Intelligent Systems
Ljubljana, Slovenia

Mitja Luštrek
mitja.lustrek@ijs.si
Jožef Stefan Institute
Department of Intelligent Systems
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia

Abstract

Waste Quantification Solutions to Limit Environmental Stress (WASTELESS) project aims to develop and test innovative tools and methodologies for measuring and monitoring food loss and waste (FLW). A key objective is to create a decision support toolbox that helps food actors across the entire supply chain, including consumers, select the most suitable method for measuring and monitoring FLW. To help with this decision, existing, already tested FLW measurement practices can be consulted, which are currently published as short documents. In this work, we show how the data about them can be extracted using large language models (LLMs). Additionally, we propose how this data can be structured and represented as an ontology. With this process, we can help users find relevant data without needing to browse through many documents.

Keywords

food loss and waste, large language models, data extraction, ontology

1 Introduction

The project *Waste Quantification Solutions to Limit Environmental Stress* (WASTELESS; https://wastelesseu.com/) is designed to develop and test a mix of innovative tools and methodologies for food loss and waste (FLW) measurement and monitoring. One of the tasks is also to create a decision support toolbox [10]. It should help all profiles of food actors, i.e. across the whole food supply chain (FSC), including consumers, who want to measure and monitor their FLW, to select the most appropriate method.

There have been several attempts to harmonise FLW measurement methods. The *Food loss and waste accounting and reporting standard* (FLW Standard; [7]) stands out as a good structured attempt. It was produced by the Food Loss & Waste Protocol, a multi-stakeholder partnership with involvement by Food and Agriculture Organization of the United Nations (FAO) and World Resources Institute among others.

The FLW standard establishes the scope of an FLW inventory. Furthermore, it provides definitions of boundary elements and recommendations for classifications that should be used to describe them. For classifying food into categories, it suggests the FAO's and World Health Organization's Codex General Standard for Food Additives [5]. We might add that alternatively, Annex II of "Regulation (EC) No 1333/2008 of the European Parliament and of the Council" can also be used. For lifecycle stage, the International Standard Industrial Classifications of All Economic Activities (ISIC) or the Statistical Classification of Economic Activities in the European Community (NACE) [4] should be used. Finally, for geographical boundary classification UN region or country codes should be used or Nomenclature of Territorial Units for Statistics (NUTS) [2] in the European context.

The FLW standard also provides guidelines on how to decide which quantification method to use for FLW measurement or monitoring. The *FLW Quantification method ranking tool* was prepared by the Waste and Resources Action Programme (WRAP) and includes eleven questions. Most of the questions serve as exclusion criteria. For example, a negative response to either "Do you have existing records that could be used for quantifying FLW?" (Q9) or "Do you have access to those records?" (Q10) excludes the method of records. As another example, a negative response to "Can you get direct access to the FLW being quantified" (Q3) immediately excludes direct weighing, counting, assessing volume, and waste composition analysis, since these all need such access to be feasible. These questions encapsulate the most important characteristics by which these methods distinguish from one another and lend themselves to particular needs of users.

In this paper, we build upon this work by proposing a unified structure through which to describe various practices of FLW measurement and reduction. This is a step towards systematic representation of these data that can enable further analysis of the practices thus described and their comparison and validation.

2 Methods

We first outline the structure of desired shortened descriptions, report on the process of using large language models (LLMs) to automatically extract them and finally evaluate the results by comparing them to human annotations.

Information Society 2025, Ljubljana, Slovenia

https://doi.org/10.70314/is.2025.skui.8598

2.1 Structure of Extracted Information

Based on the previously mentioned *FLW Quantification method ranking tool* and domain-expert knowledge, we determined the following characteristics of FLW measurement methods and practices to be of the most importance:

(1) FLW method.
FLW measurement and reduction practices might describe very specific technologies and techniques. To make the information more general, we decided to classify as one of ten categories of quantification methods. These are described in detail in the Supplement [8] to the *FLW Accounting and Reporting Standard* [7].

(2) Region of interest.
European Union (EU) member countries have diverse legislation that is of relevance to FLW measurement (see [13] for a review). Some have legislation actions that are legally binding, such as laws and regulations, and as such prescribe methods of monitoring and FLW measurement as well as the ways of reporting the data. On the other hand, some countries only approach the topic through non-binding legislation actions, such as agency orders and policy papers. As such, not every method might be appropriate for every country or region.

(3) Food supply chain (FSC) stage.
Food loss and waste can occur at any stage of the food supply chain, starting from farmers and other producers, through food manufacturers and processors, distributors and shippers, grocery stores and restaurants, all the way to the customers and consumers. Some methods are more appropriate for certain stages in this chain. For example, a household might keep a diary of their FLW, while sellers such as grocery stores, generally manage their stock more systematically and precisely.

(4) Accuracy.
FLW measurement methods need also to be considered from the point of desired accuracy. The highest accuracy can be achieved by directly weighing the waste or separating it into components (waste composition analysis), while diaries or volume assessment produce data of medium accuracy. At the lowest end, proxy data can be used to assess FLW, for example by using data from another region to extrapolate findings to another; keeping in mind that such data will not be very accurate.

(5) Food category.
Depending on the type of food and how it is packed, we might only be able to use some FLW measurement methods, but not others. For example, when dealing with packed food items, wasted products can be simply counted and their weight inferred. Meanwhile, when waste occurs with liquid food, such as milk, volume assessment can be fairly accurate to estimate the weight of FLW.

(6) Direct access to FLW.
Some food waste cannot be measured directly, such as by weighing, counting, or waste composition analysis. For example, when waste is discarded directly into the drain in the process of food processing, it might be mixed with other waste water exiting the processing plant. In cases like this, non-direct methods need to be employed, such as modelling or mass balance.

To be able to suggest specific FLW practices according to the criteria described above, we need to first describe them in terms of these characteristics. For harmonious representation, we used already mentioned NUTS and NACE classifications for region of interest and FSC stage, respectively. We also used a simplified version of FAO's *Global individual food consumption* (GIFT; [6]) classification to describe food category. For accuracy, we opted for three categorical levels of "low", "medium", and "high", while direct access to FLW can be represented with a simple Boolean.

2.2 Extraction of Data

To test the extraction of data, we used 11 FLW measurement and reduction practice descriptions. This included 3 descriptions of practices developed and piloted in the WASTELESS project as well as 8 practices developed in other European projects [16].

To extract data from FLW practice descriptions, we used two LLMs: ChatGPT 5 Auto [12] and Le Chat [11]. The prompt consisted of the following:

(1) Introduction: general summary of the whole extraction process;
(2) Main instructions:
  (a) Information to be extracted: a list of questions, the answers to which represent the data that is to be extracted from the practice description;
  (b) Data types and values: a list of possible values and their types for each of the data field, including lists of NUTS and NACE codes and food categories;
  (c) Missing information: instructions on how to deal with missing, incomplete, or unclear data;
  (d) Format: description of the format of expected output (`.csv` data);
(3) Example:
  (a) Input: a short, synthetic description of a FLW practice;
  (b) Reasoning: values for all data fields and their relationship to original text, indicate missing values;
  (c) Output: the expected line of data output.

We included all reference classifications as `.csv` files as well as the *Guidance on FLW Quantification Methods* as a PDF.

Following this initial prompt, practice descriptions were uploaded one by one and the output saved. The lead author of this paper also extracted the same information from the descriptions manually.

2.3 Evaluation of Results

To evaluate the extraction of data by LLMs, we compared the output by these models to human annotations. Here, the cases of multiple possible values and missing data need to be considered. First, some characteristics can objectively contain several values. For example, a FLW measurement practice might be applicable to several FSC stages and more than one food category. Secondly, some data cannot be determined from the description of practice.

For a characteristic with more than one possible value, consider two subsets of all possible values (U): human annotations (H) and machine-extracted values (M). The following list gives the scores that were used in the evaluation for all possible relationships of these two sets.

+2; when the subsets were equal, $H = M$.
+1; when an LLM extracted more values than a human, but including those, $\emptyset \neq H \subset M \neq U$.
0; when the sets were overlapping, but neither contained the other, that is, there was a partial match in values, $H \cap M \neq \emptyset, H \not\subseteq M, M \not\subseteq H$.
0; when there was data available, but LLM extracted no information or returned all possible values, $\emptyset \neq H \subset U$, but $M = \emptyset$ or $M = U$.
-1; when an LLM failed to extract all values that a human did, $U \supseteq H \supset M \neq \emptyset$.
-2; when the subsets had no values in common, i.e., were disjoint, $H \cap M = \emptyset$.

Note that for simple true or false values, this list simplifies to the extreme cases; thus they were scored as +2 and −2, respectively.

The reasoning behind the scoring is that we prefer to describe a practice in broader terms, even if some extracted values are inapplicable, rather than miss a particular value. As an example, it is better to describe a practice as suitable for all food categories than missing the one that it is actually suitable for. Similarly, when no information is extracted, we can conservatively assume all values apply. In such a case, an LLM failed objectively, but it is not punished for it. In the worst case scenario, an LLM "extracted" or hallucinated some values, but they have nothing in common with human annotations; for this two points are deducted.

3 Results

To evaluate the extraction of data by the LLMs, we scored their answers as described in Section 2.3. We summarised these scores for each practice characteristic in Table 1, where shown are the sum of scores and the number of perfect scores, that is the number of times the LLM completely agreed with the human rater. The number of practices tested was 11, which is therefore the maximum number of perfect scores, while the maximum sum is 22.

Table 1: Agreement scores for each characteristic of a FLW practice between a human rater and two different LLMs. The sum of scores and the number of perfect extractions are shown.

Model	ChatGPT		Le Chat	
Metric	Sum	Perfect	Sum	Perfect
FLW method	13	8	3	5
Region	12	7	13	5
FSC stage	8	7	12	6
Accuracy	−2	4	−5	3
Food category	22	11	21	10
Direct access	6	7	14	9
Total	59	44	58	38

Both models achieved similar scores in total across all practice characteristics. ChatGPT did, however, perfectly agree with the human rating more often. Of all the characteristics, food category was the easiest for the LLMs to extract. This is a simple classification and usually, the type of food is mentioned explicitly. The FLW quantification method was inferred with moderate success. On the other hand, accuracy of methods was very poorly described.

4 Discussion

In this work, we have shown how using two LLMs, the data from unstructured FLW measurement and reduction practice descriptions can be extracted into structured data. We achieved satisfying if imperfect results.

The most important data point, which is the class of the FLW measurement method was extracted with moderate success. It needs to be pointed out that extracted information was not wildly inaccurate in most cases, despite of what the scores might suggest. For example, a method of tracking waste on a blockchain was classified as using records, where in fact, the data were collected with surveys before being, indeed, *recorded*. Similarly, one practice described weighing waste as it was collected in the wastebasket, while simultaneously taking photos of the material. Here, the true measurement method was direct weighing, but the LLMs classified it as waste composition analysis. By using photos, such an analysis could in theory be done, but was not in such case. Thus, to improve the relevance of the FLW measurement method, we might instead group them by some other characteristics. For example, we could drop the data field of direct access and instead consider groups of methods separated in terms of needing direct access to waste.

Food category, however, was very reliably extracted. This indicates that in the further process of the extracted data, we could make the best use of the food type. Accuracy of the method described was not extracted well, but this is most likely due to the subjectivity of this characteristic. The authors of FLW practice descriptions never explicitly addressed the question of accuracy, so it needed to be estimated roughly by other characteristics, such as the general accuracy of the FLW method class. This also suggests that a three-level accuracy is probably too fine grained and it should be described only as "low" and "high".

We should note that our evaluation only compares the performance of LLMs to manual extraction of data performed by a single person. It is expected that people would also differ in their extractions, i.e., would not achieve perfect inter-rater agreement. Thus, the evaluation should not be interpreted as how well the LLMs captured the "objective" truth.

With this process, LLMs enabled us to transform the descriptions from simple PDF files into structured CSV files in a semi-automatic way. In terms of the five-star rating of open data [9] which describes how to get from data in proprietary formats to linked open data, we thus increase their level from one star to three stars. We can extend this further and increase the rating of this data to five stars: publish truly linked data.

The first step that can follow directly the results of this work is to transform the structure described in Section 2.1

Listing 1: A snippet of the ontology in Turtle language [1]

```
@prefix : <http://purl.archive.org/fwo/> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix dbpedia: <http://dbpedia.org/resource/> .
@base <http://purl.archive.org/fwo/> .

<http://purl.archive.org/fwo> rdf:type owl:Ontology .

#####################  Classes  #####################
:FoodLossWasteMeasurementPractice rdf:type owl:Class ;
     rdfs:label "Food Loss and Waste Measurement
          Practice"@en .

:Region rdf:type owl:Class ;
        rdfs:label "A NUTS code of the region" ;
        owl:equivalentClass
             dbpedia:Nomenclature_of_Territorial_Units-
        _for_Statistics .

:FoodCategory rdf:type owl:Class ;
              rdfs:label "Food Category" .

:DairyAndEggs rdf:type owl:Class ;
              rdfs:subClassOf :FoodCategory ;
              rdfs:label "Dairy & Eggs" .

:Milk rdf:type owl:Class ;
      rdfs:subClassOf :DairyAndEggs ;
      rdfs:label "Milk" .
# ... more classes defined ...
################  Object Properties  ################
:hasTitle rdf:type owl:DatatypeProperty ;
    rdfs:domain :FoodLossWasteMeasurementPractice ;
    rdfs:range rdfs:Literal ;
    rdfs:label "with the title" .

:hasRegion rdf:type owl:ObjectProperty ;
    rdfs:domain :FoodLossWasteMeasurementPractice ;
    rdfs:range :Region ;
    rdfs:label "applied in regions" .

:hasFoodCategory rdf:type owl:ObjectProperty ;
        rdfs:domain :FoodLossWasteMeasurementPractice ;
        rdfs:range :FoodCategory ;
        rdfs:label "applicable to food categories" .

:hasAccuracy rdf:type owl:DatatypeProperty ;
    rdfs:domain :FoodLossWasteMeasurementPractice ;
    rdfs:range "low"^^xsd:string,
         "medium"^^xsd:string, "high"^^xsd:string .
```

into an ontology. We illustrate this idea in Listing 1 which encodes the characteristics as classes and how to connect these to an individual practice using object and datatype properties. Once we represent the structure like this, we can encode a specific instance of FLW measurement practice as:

```
:MyDairyWastePractice a
     :FoodLossWasteMeasurementPractice ;
    :hasTitle "Tracking Waste of Dairy in Slovenia" ;
    :hasFoodCategory :WholeMilk ;
    :hasAccuracy "high"^^xsd:string ;
    :hasRegion :SI0.
```

The data on FLW measurement practices can then be easily linked to other published data and the closest candidate ontology is the *Food Waste Ontology* by Stojanov et al. [15]. The dataset described by this ontology is already vast and is being extended through *FoodWasteEXplorer* [14]. By leveraging it, we plan to publish the practice descriptions as five-star data in future work.

Acknowledgments

This work was carried out as a part of the WASTELESS project which is funded by the European Union's Horizon Europe Research and Innovation programme under Grant Agreement No. 101084222.

References

[1] David Beckett, Tim Berners-Lee, Eric Prud'hommeaux, and Gavin Carothers. 2014. *RDF 1.1 Turtle. Terse RDF Triple Language*. Ed. by Eric Prud'hommeaux and Gavin Carothers. World Wide Web Consortium (W3C), (Feb. 25, 2014). Retrieved Aug. 29, 2025 from https://www.w3.org/TR/turtle/.

[2] European Parliament and Council of the European Union. 2003. Regulation (EC) no 1059/2003 of the European Parliament and of the Council. On the establishment of a common classification of territorial units for statistics (NUTS). *Official Journal of the European Union*, 154, 1, (June 21, 2003), 1–41. http://data.europa.eu/eli/reg/2003/1059/oj.

[3] European Parliament and Council of the European Union. 2008. Regulation (EC) no 1333/2008 of the European Parliament and of the Council. On food additives. Version 02008R1333-20240423. *Official Journal of the European Union*, 354, 16, (Dec. 16, 2008), 16–33. http://data.europa.eu/eli/reg/2008/1333/oj.

[4] European Parliament and Council of the European Union. 2006. Regulation (EC) no 1893/2006 of the European Parliament and of the Council. Establishing the statistical classification of economic activities NACE revision 2 and amending Council Regulation (EEC) no 3037/90 as well as certain EC regulations on specific statistical domains. *Official Journal of the European Union*, 393, (Dec. 20, 2006), 1–39, 1, (Dec. 20, 2006). http://data.europa.eu/eli/reg/2006/1893/oj.

[5] Food and Agriculture Organization of the United Nations and World Health Organization. 2019. General standard for food additives. Codex STAN 192-1995. (2019).

[6] Food and Agriculture Organization of the United Nations (FAO). 2022. *FAO/WHO GIFT. Global Individual Food Consumption Data Tool*. Retrieved Aug. 30, 2025 from https://www.fao.org/gift-individual-food-consumption/about/en.

[7] Craig Hanson et al. 2016. *Food Loss and Waste Accountingand Reporting Standard*. Version 1.0. World Resources Institute. ISBN: 978-1-56973-892-4.

[8] Craig Hanson et al. 2016. *Guidance on FLW Quantification Methods. Supplement to the Food Loss and Waste (FLW) Accounting and Reporting Standard*. World Resources Institute. ISBN: 978-1-56973-893-1.

[9] Tim Berners Lee. 2006. Linked data. Design issues. Version 2009-06-18. (July 27, 2006). https://www.w3.org/DesignIssues/LinkedData.html.

[10] Mitja Luštrek and Junoš Lukan. 2024. Practice Abstracts – batch 1 – early phase. Deliverable 6.2. Research rep. Jožef Stefan Institute. doi:10.5281/ZENODO.13503261.

[11] [SW] Mistral AI, Le Chat version November 2024, 2024. url: https://chat.mistral.ai/.

[12] [SW] OpenAI, ChatGPT version GPT-5, 2025. url: https://chatgpt.com/.

[13] Zhuang Qian, Wu Chen, and Giorgia Sabbatini. 2023. White book for FLW reduction, measurement, and monitoring practices. Deliverable 1.1. Research rep. Version 1.0. University of Southern Denmark, (Aug. 30, 2023). 116 pp. doi:10.5281/ZENODO.11065358.

[14] REFRESH. FoodWasteEXplorer. https://www.foodwasteexplorer.eu/about.

[15] Riste Stojanov, Tome Eftimov, Hannah Pinchen, Maria Traka, Paul Finglas, Drago Torkar, and Barbara Korousic Seljak. 2019. Food waste ontology. A formal description of knowledge from the domain of food waste. In *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, (Dec. 2019). doi:10.1109/bigdata47090.2019.9006254.

[16] Sustainable Food System Innovation Platform. Practice abstract inventory. Retrieved Sept. 1, 2025 from https://www.smartchain-platform.eu/en/practice-abstract-inventory.

Eye-Tracking Explains Cognitive Test Performance in Schizophrenia

Mila Marinković
Jure Žabkar
mila.marinkovic@fri.uni-lj.si
jure.zabkar@fri.uni-lj.si
University of Ljubljana,Faculty of Computer and Information Science, Ljubljana, Slovenia

Abstract

Schizophrenia is associated with cognitive impairments that are difficult to assess with traditional neuropsychological tests, which are often lengthy and burdensome. Eye-tracking (ET) provides objective, minimally invasive measures of visual attention and cognitive processing and may complement shorter assessments. This study investigated whether ET features recorded during three computerized tasks could distinguish patients with schizophrenia from healthy controls. Using the Explainable Boosting Machine (EBM), we achieved an accuracy of 0.86, and balanced sensitivity and specificity, with an area under the curve exceeding 0.9. Features related to fixation patterns, saccadic dynamics, and temporal engagement emerged as the most informative. These findings indicate that ET features collected during brief cognitive tasks can provide clinically relevant markers of schizophrenia. Incorporating ET into short test batteries may reduce patient burden while enhancing diagnostic value, supporting the development of scalable and practical screening tools.

Keywords

schizophrenia, eye-tracking, cognitive tasks, machine learning

1 Introduction

Schizophrenia is a severe and chronic neuropsychiatric disorder that affects about 1% of the population worldwide and is characterized by disturbances in thought, perception, and behavior [1]. In addition to positive and negative symptoms, patients experience pronounced cognitive impairments, including deficits in attention, working memory, and executive functioning, which substantially affect everyday life outcomes [2, 3]. Cognitive assessment is therefore central for both diagnosis and monitoring of schizophrenia. However, traditional neuropsychological testing is lengthy, cognitively demanding, and often exhausting for patients, limiting its feasibility in clinical practice. Shorter test batteries reduce the burden but often fail to provide sufficiently informative data for reliable diagnosis

Eye-tracking (ET) offers a promising avenue for addressing this challenge. ET provides objective, real-time measures of visual attention, oculomotor control, and information processing strategies [4]. Numerous studies have shown that patients with schizophrenia exhibit abnormalities in smooth pursuit eye movements, antisaccades, and fixation stability [5, 6, 7]. These alterations are considered potential endophenotypes of the disorder, as they are also observed in first-degree relatives who do not have schizophrenia [6, 7]. More recent work has extended ET beyond basic oculomotor paradigms by embedding it in cognitive tasks. For example, Okazaki et al. [8] combined ET metrics with digit-symbol substitution tests and showed improved discrimination between patients and controls. Yang et al. [9] reported that abnormal gaze patterns during reading tasks—such as longer fixation durations and increased saccade counts—enabled high diagnostic accuracy when analyzed with machine learning models. Similarly, Morita et al. [10] demonstrated the feasibility of portable tablet-based ET combined with cognitive assessments for schizophrenia screening. Collectively, these studies highlight that combining ET with cognitive testing enriches diagnostic value and provides insights into the cognitive mechanisms underlying gaze abnormalities.

Building on this prior work, the present study investigates whether ET features recorded during a small set of computerized cognitive tasks can serve as reliable markers of schizophrenia. Participants completed three tasks (digit span, picture naming, and n-back), each divided into phases of instruction reading, video demonstration, and test execution. From these tasks, we extracted 117 ET features, including fixation measures, saccadic dynamics, gaze entropy, and recording duration. We then applied machine learning methods to evaluate the discriminative power of these features. By focusing on only three short tasks, our aim is to test whether ET provides sufficient additional information to overcome the limitations of brief cognitive testing, ultimately supporting the development of less burdensome but more informative screening approaches.

2 Methods

2.1 Participants

The study involves 126 individuals, including 58 patients diagnosed with schizophrenia (SP) and 68 healthy controls (HC). All participants were adults, aged 18 years or older. Patients were recruited and tested at the University Psychiatric Hospital Ljubljana. The control group was matched to the patient group on age and gender.

Eligibility criteria required fluency in Slovenian and excluded individuals with intellectual disability, organic brain disorders, or a history of substance abuse. Additional exclusion criteria for the HC group included any past or current psychiatric disorder. At the time of assessment, all SP participants were receiving stable doses of antipsychotic medication.

Demographic characteristics of the two groups are presented in Table 1 and were analyzed to ensure that the groups were comparable in terms of age and gender. While educational attainment differed between groups, further analyses confirmed that within each education level there were no significant differences between SP and HC participants, indicating that education was unlikely to confound the comparisons.

Information Society 2025, Ljubljana, Slovenia

https://doi.org/10.70314/is.2025.skui.9583

Table 1: Demographic characteristics of participants.

Measure	SP	HC
Counts		
Total participants	58	68
Male sex	29	35
Female sex	29	33
Continuous		
Age (mean years)	46.1	46.7
Categorical		
Most common education level	Primary school	High school

HC: Healthy Controls; SP: Patients with Schizophrenia

The study was approved by the Medical Ethics Committee of the Republic of Slovenia (approval number: 0120-51/2024-2711-4). All participants received a detailed explanation of the study procedures and provided written informed consent prior to participation.

2.2 Testing Procedure

Eye-tracking data were collected using a Tobii Pro Spectrum [11] eye tracker integrated into a 24-inch monitor with a resolution of 1920×1080 pixels. Recordings were made at 1200 Hz in the "human" tracking mode, with a stimulus presentation latency of approximately 10 ms. The display frame rate was 30 FPS. Participants sat ~55cm from the monitor, in a upright position with seating adjusted for comfort and optimal tracking.

Before each task, participants were seated comfortably, and the Tobii Pro Lab [12] interface provided a live preview (see Fig. 1) to verify that both eyes were detected and that the viewing distance was within the recommended range (displayed as a green zone, typically around 55 cm). Once this was confirmed, a standard five-point calibration was performed, during which participants followed a moving dot across the screen. Calibration served both to align gaze tracking and to ensure that the participant had not moved their head between tasks. If the system indicated suboptimal accuracy, the calibration was repeated.

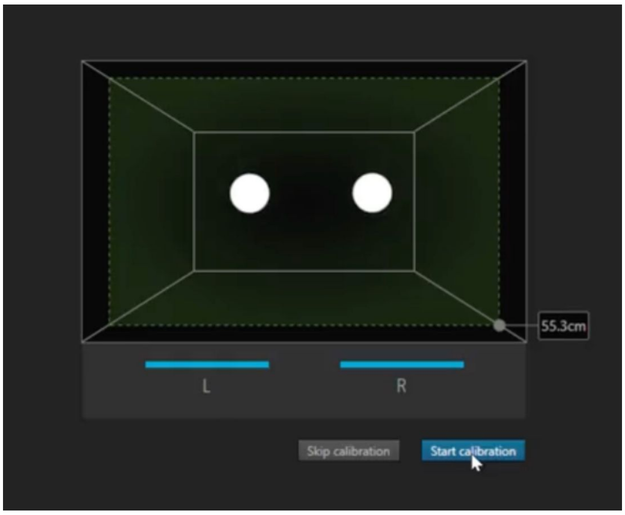


Figure 1: Calibration interface in Tobii Pro Lab. The preview window ensures both eyes are detected and the participant is seated at an appropriate distance (green zone, approximately 55 cm) before calibration and testing begin.

Participants completed three computerized cognitive tasks in a fixed order: digit span (DS), n-back (NB), and picture naming (PS). A short break was provided between tasks, with the duration determined by each participant. All tasks were presented within the Tobii Pro Lab application, which also stored the raw data. After recording, the data were exported and processed using a custom Python program for feature extraction and analysis.

Each task followed the same three-phase structure:

(1) **Reading instructions.** Written instructions were displayed on the screen. Participants could read them at their own pace and advanced to the next phase with a mouse click.
(2) **Video example.** A short instructional video was presented once, demonstrating the task procedure.
(3) **Test execution.** The participant began the task when ready. Task duration depended on individual performance.

The procedure was identical for all participants, ensuring standardization across groups. Only the test execution phase varied in length, as it was determined by each participant's performance. Group-level descriptive statistics of fixation durations for all tasks and phases are reported in the Results section (Table 3).

2.3 Feature Extraction

We extracted a total of 117 ET features from three computerized cognitive tasks. As described in Section 2.2, each task was divided into three phases: instruction reading (BN), video demonstration (GN), and test execution (T).

Each participant contributed a single data point to the ML analysis. For every task (DS, PS, NB) and every phase (BN, GN, T), we computed the 13 eye-tracking features listed in Table 2. Each feature was calculated over the entire duration of the given phase (e.g., the number of fixations refers to the total count during that phase, while mean fixation duration refers to the average across all fixations in that phase). These were then concatenated across all tasks and phases, yielding 117 features per participant. Thus, the unit of analysis was the participant, not individual trials or task phases.

2.4 Data Analysis

We trained and evaluated several machine learning models using these features. We applied stratified 10-fold cross-validation at the subject level to ensure that all features from a given participant were assigned exclusively to either the training or test set, thereby preventing data leakage across folds. In each iteration, the model was trained on nine folds and tested on the remaining one, and the reported metrics represent averages across all folds. Performance was assessed using accuracy, sensitivity, specificity, and the area under the receiver operating characteristic curve (AUC). The final results were reported as the average across all folds.

We evaluated a diverse set of ML models (logistic regression, random forest, gradient boosting, extreme gradient boosting, and the explainable boosting machine) to cover both linear and non-linear approaches with varying levels of interpretability. EBM was selected as the primary model because it consistently achieved the highest overall performance while providing inherently interpretable feature importance, which is particularly valuable in clinical contexts. We did not pursue deep neural networks in this study, as the dataset size (126 participants) is relatively small and does not provide sufficient power to train high-capacity models without overfitting.

Table 2: Eye-tracking features extracted from each task and phase.

Feature	Description
`num_fixations`	Total number of fixations during the interval.
`avg_fixation_duration`	Mean duration of fixations (ms), indicating fixation stability.
`std_fixation_duration`	Standard deviation of fixation duration, reflecting variability in fixation times.
`num_saccades`	Total number of saccadic eye movements.
`avg_saccade_distance`	Mean distance of saccades, reflecting amplitude of eye shifts.
`avg_saccade_velocity`	Mean velocity of saccades, indicating how quickly gaze shifts occurred.
`avg_saccade_angle`	Average angular change of saccades, reflecting directional scanning patterns.
`gaze_entropy`	Entropy of gaze distribution, quantifying dispersion vs. concentration of gaze.
`recording_duration_ms`	Total duration of recording for the phase (ms).
`unique_squares`	Number of unique spatial areas (AOIs) visited during the interval.
`num_changes`	Number of transitions between distinct gaze areas.
`missing_left_percent`	Percentage of missing data from the left eye.
`missing_right_percent`	Percentage of missing data from the right eye.

Note: All features are computed as aggregates over the entire task phase for each participant.

3 Results

To characterize task engagement and potential variability between groups, we compared fixation durations across all tasks and phases (Table 3). SP showed longer fixations than HC, especially during instruction reading and video phases, with smaller but consistent differences during execution. This indicates altered attention even outside active task solving.

Table 3: Mean fixation duration in ms per task and phase.

Task	Phase	HC (Mean ± SD)	SP (Mean ± SD)
Numbers	Reading instructions	239.64 ± 47.79	283.97 ± 45.33
	Watching video	352.14 ± 81.56	400.10 ± 89.51
	Test execution	390.66 ± 83.92	407.60 ± 98.53
Pictures	Reading instructions	228.44 ± 52.49	267.78 ± 60.79
	Watching video	302.40 ± 69.06	368.93 ± 81.42
	Test execution	301.97 ± 49.91	319.36 ± 58.07
Square	Reading instructions	229.36 ± 45.41	286.70 ± 63.42
	Watching video	309.41 ± 89.45	352.08 ± 79.37
	Test execution	394.91 ± 115.50	406.24 ± 99.36

SD: Standard deviation; HC: Healthy controls; SP: Schizophrenia patients

The ML models were trained on 117 extracted eye-tracking features and achieved strong performance in distinguishing SP from HC. The key cross-validation performance metrics are summarized in Table 4.

Table 4: Cross-validation performance metrics for different models. The Explainable Boosting Machine (EBM) achieved the best overall performance across all metrics.

Model	Accuracy	Sensitivity	Specificity	AUC
EBM	**0.86**	**0.84**	**0.86**	**0.93**
LR	0.85	0.77	0.91	0.92
GB	0.78	0.70	0.84	0.83
RF	0.83	0.84	0.82	0.91
xGB	0.81	0.77	0.85	0.90

EBM: Explainable Boosting Machine; LR: Logistic Regression; GB: Gradient Boosting; RF: Random Forest; xGB: Extreme Gradient Boosting

Among the tested models, the EBM achieved the highest overall performance and was therefore selected for detailed analysis. Fig 2 presents the receiver operating characteristic (ROC) curve, which confirms the model's strong discriminative ability.

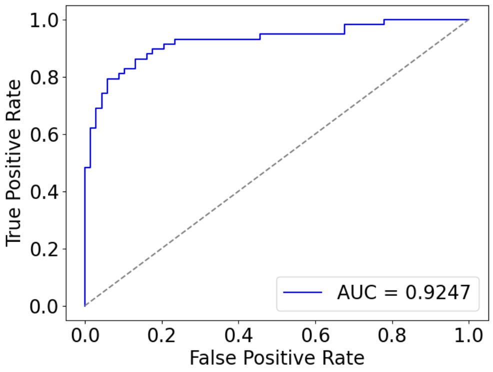


Figure 2: ROC curve for the EBM model. The mean AUC across folds was 0.92, confirming strong classification performance.

We analyzed the feature importance scores provided by EBM, focusing on the ten most informative features (Fig 3). These features were predominantly derived from the test execution phases and included measures such as recording duration, number of fixations, mean fixation duration, and saccadic counts.

4 Discussion

The present study demonstrates that eye-tracking (ET) features obtained during brief computerized cognitive tasks can effectively discriminate between individuals with schizophrenia and healthy controls. Using 117 features, the Explainable Boosting Machine (EBM) achieved strong classification performance, with accuracy, sensitivity, and specificity values around 0.85 and an AUC of 0.92. These results provide further evidence that ET-based measures capture clinically relevant differences in cognitive processing and attentional control in schizophrenia.

Our findings are consistent with previous work showing that patients with schizophrenia exhibit abnormalities in fixation behavior, saccadic dynamics, and gaze distribution during

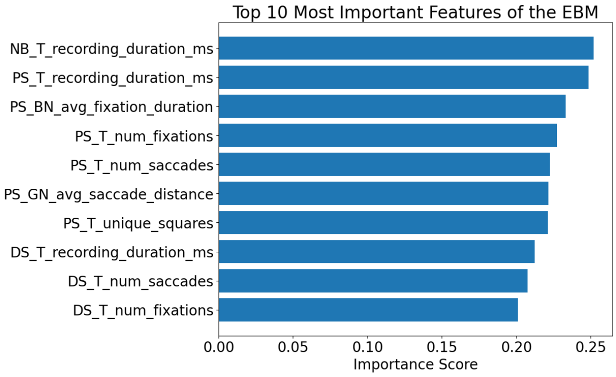


Figure 3: Top 10 most important features identified by the EBM model. The prefixes indicate the task and phase: DS = digit span, PS = picture naming, NB = n-back; BN = reading instructions, GN = watching video, T = test solving. For example, `PS_T_num_fixations` refers to the number of fixations during the test phase of the picture naming task.

both simple oculomotor paradigms and more complex cognitive tasks [5, 6, 7, 8, 9, 10]. Importantly, by embedding ET into a small set of standardized cognitive tasks, we demonstrate that group differences emerge not only during active problem solving but also in more passive phases such as reading instructions or watching a video example. This suggests that ET provides valuable information across the continuum of cognitive engagement, extending beyond traditional task performance metrics.

While prior studies have applied machine learning to ET data in schizophrenia, they have typically relied on single paradigms or isolated task conditions. The novelty of the present work lies in combining a multi-task, multi-phase design with interpretable ML within a short, clinically feasible test battery. This approach captures a broader range of cognitive and attentional processes while linking model performance to specific, clinically meaningful features.

Interpretability showed that temporal engagement, fixation stability, and saccadic activity best differentiated groups. Longer recording durations may reflect slower processing, while altered fixations and saccades align with prior reports of impaired attentional control. These findings suggest that eye-tracking captures both temporal and oculomotor aspects of task performance, supporting its potential as a clinically meaningful biomarker.

From a clinical perspective, these results are encouraging. Traditional neuropsychological assessments are lengthy and cognitively demanding, which can be exhausting for patients and limit their applicability. Our study shows that by integrating ET measures into just three relatively brief cognitive tasks, it is possible to achieve a high level of diagnostic accuracy. This approach may therefore support the development of shorter, less burdensome, and more objective screening protocols that could complement existing clinical evaluations.

Limitations and Future Work

Several limitations should be noted. First, although our sample size of 126 participants is comparable to similar studies, larger and more diverse cohorts are needed to confirm the generalizability of the results. Second, all patients were on stable antipsychotic medication, which may have influenced oculomotor behavior. Third, while we employed subject-level cross-validation to prevent data leakage, robustness checks such as leave-one-subject-out or leave-one-task-out validation could further strengthen reliability. Fourth, our analysis focused on static ET features; dynamic sequence-based or deep learning models could capture additional temporal information in gaze patterns. Finally, we only tested three tasks; future research should explore whether expanding or tailoring the task battery improves performance while still keeping the protocol brief. Replication with independent cohorts will be essential to establish clinical utility.

Conclusion

In conclusion, this study provides strong evidence that eye-tracking features embedded within short cognitive tasks can serve as robust markers of schizophrenia. Machine learning models trained on these features achieved high discriminative accuracy, with interpretable patterns that align with known attentional and cognitive impairments in the disorder. By reducing patient burden while maintaining informativeness, this approach holds promise for the development of accessible, scalable, and clinically relevant screening tools for schizophrenia.

5 Acknowledgments

This research was funded by the Slovenian Research Agency (core funding No. P2-0209). Additional support for Mila Marinković was provided through the ARIS project AI4Science (GC-0001). We thank the University Psychiatric Hospital Ljubljana for participant recruitment, asist. dr. Polona Rus Prelog, dr. med., spec. psih., and Martina Zakšek for their support, and the broader research team, clinicians, and participants for their contributions. The authors declare that they have no conflict of interest.

References

[1] S. R. Marder and T. D. Cannon, "Schizophrenia," *The New England Journal of Medicine*, vol. 381, no. 18, p. 1753–1761, 2019.

[2] W. Hinzen and J. Rosselló, "The linguistics of schizophrenia: thought disturbance as language pathology across positive symptoms," *Frontiers in Psychology*, vol. 6, 2015.

[3] L. Colle, R. Angeleri, M. Vallana, K. Sacco, B. Bara, and F. Bosco, "Understanding the communicative impairments in schizophrenia: A preliminary study," *Journal of Communication Disorders*, vol. 46, no. 3, pp. 294–308, 2013.

[4] A. Wolf, K. Ueda, and Y. Hirano, "Recent updates of eye movement abnormalities in patients with schizophrenia: A scoping review," *Psychiatry and Clinical Neurosciences*, vol. 75, pp. 104–118, 2021.

[5] P. S. Holzman, L. R. Proctor, and D. W. Hughes, "Eye-tracking patterns in schizophrenia," *Science*, vol. 181, no. 4095, pp. 179–181, 1973.

[6] L. Deborah, H. Philip, M. Steven, and M. Nancy, "Eye tracking and schizophrenia: A selective review," *Schizophrenia Bulletin*, vol. 20, no. 1, pp. 47–62, 1994.

[7] U. Ettinger, "Smooth pursuit and antisaccade eye movements as endophenotypes in schizophrenia spectrum research," PhD Thesis, Department of Psychology, Goldsmiths College, University of London, 2002.

[8] K. Okazaki, K. Miura, J. Matsumoto, N. Hasegawa, M. Fujimoto, H. Yamamori, Y. Yasuda, M. Makinodan, and R. Hashimoto, "Discrimination in the clinical diagnosis between patients with schizophrenia and healthy controls using eye movement and cognitive functions," *Psychiatry and Clinical Neurosciences*, vol. 77, pp. 393–400, 2023.

[9] H. Yang, L. He, W. Li, Q. Zheng, Y. Li, X. Zheng, and J. Zhang, "An automatic detection method for schizophrenia based on abnormal eye movements in reading tasks," *Expert Systems With Applications*, vol. 238, p. 121850, 2024.

[10] K. Morita, K. Miura, A. Toyomaki, M. Makinodan, K. Ohi, N. Hashimoto, Y. Yasuda, T. Mitsudo, F. Higuchi, S. Numata, A. Yamada, Y. Aoki, H. Honda, R. Mizui, M. Honda, D. Fujikane, J. Matsumoto, N. Hasegawa, S. Ito, H. Akiyama, T. Onitsuka, Y. Satomura, K. Kasai, and R. Hashimoto, "Tablet-based cognitive and eye movement measures as accessible tools for schizophrenia assessment: Multisite usability study," *JMIR Mental Health*, vol. 11, p. e56668, 2024.

[11] M. Nyström, D. Niehorster, R. Andersson, and I. Hooge, "The tobii pro spectrum: A useful tool for studying microsaccades?" *Behavior Research Methods*, vol. 53, 07 2020.

[12] Tobii AB, "Tobii pro lab," Computer software, Danderyd, Stockholm, 2024. [Online]. Available: http://www.tobii.com/

Data-Driven Evaluation of Truck Driving Performance with Statistical and Machine Learning Methods

Vid Nemec
vidotti.nemec@gmail.com
Jožef Stefan Institute
Ljubljana, Slovenia

Gašper Slapničar
Jožef Stefan Institute
Ljubljana, Slovenia
gasper.slapnicar@ijs.si

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

Figure 1: Truck driving simulator developed by AAER Research d.o.o.

Abstract

This paper investigates which driving features (e.g. speed, acceleration, braking) most strongly affect driving efficiency in a truck simulator environment. The work systematically compares statistical methods (thresholding based on percentiles, IQRs, expert rules) with machine learning methods (clustering using K-means) for driver assessment. In addition to practical machine learning experimentation, the analysis incorporates expert knowledge and insights from recent research. This approach evaluates the agreement and differences between the two approaches and aims to interpret them.

Keywords

Driving simulation, fuel efficiency, percentiles, K-Means, SHAP, statistical thresholds, machine learning, clustering

1 Introduction

Reducing fuel consumption in road transport is a critical goal for sustainability and cost-efficiency [1]. Prior research, such as [2, 3], highlights the impact of driver behaviour - particularly acceleration, braking, and speed profiles on overall fuel efficiency. Yet, how to most effectively quantify and compare drivers remains an open question [4]. This paper addresses which driving features most strongly influence efficiency in a simulated truck driving environment, comparing classical statistical thresholding, based on expert knowledge, with clustering - based machine learning. Applying known methods, we test whether unsupervised ML can identify driver features with stronger influence on fuel consumption than fixed-threshold rules, providing a data-driven baseline for future model-based feedback.

In addition, we compare the empirical outcomes of our ML analytics with insights from recent literature and the practical judgement of a driving expert, to pinpoint where domain knowledge aligns or conflicts with the models. This dual perspective enables a richer interpretation of driver assessment tools and informs the design of future vehicle feedback and incentive systems.

2 Related Work

Recent studies have evaluated driver behaviour for fuel efficiency using both statistical rules and machine-learning approaches. Sullivan et al. present a TORCS-based simulator with a realistic fuel-economy model, enabling safe, repeatable analysis of eco-driving strategies [5]. Maisonneuve characterises driver energy efficiency across driving events and proposes a grading/ranking method based on identified parameters [6]. Zhao et al. develop a simulator-based eco-driving support system with real-time feedback and post-drive reports, demonstrating measurable reductions in fuel use and emissions [7]. Ma et al. provide a scoping review of energy-efficient driving behaviours and applied AI methods [8]. Prototype driver-training systems have been proposed [9], and large-scale, data-driven frameworks to incentivise efficient driving have been developed [3, 10].

Most studies agree that key features include speed, throttle, brake usage, and sometimes gear selection, but differ on methods for quantifying and weighting these features. Machine learning clustering (e.g., K-means) and feature importance analysis (e.g., SHAP) are increasingly used, offering potential improvements in objectivity and interpretability of drivers.

Information Society 2025, Ljubljana, Slovenia

https://doi.org/DOI10.70314/is.2025.skui.4765

3 Methods

3.1 Data Collection and Preprocessing

Driving data were collected from a high-fidelity truck simulator developed by AAER Research d.o.o., which continuously recorded multiple parameters including pedal positions, steering wheel angle, vehicle speed, location, and segment identifiers. To ensure data quality, missing or zeroed pedal values were imputed. The signals were then resampled into 1-second windows, where for each parameter we computed the minimum, maximum, mean, and median values. This aggregation approach was chosen over raw resampling because the signals are irregular, zero-inflated, and not normally distributed, making window-based statistics more representative of driver behavior. In addition, the last observed cumulative distance within each window was retained to preserve distance continuity. Finally, the processed signals were aligned with the boundary of the scenario segment, allowing a consistent basis for later efficiency evaluation.

3.2 Rule-based Aggregation of Segment Labels

We aggregated per-segment labels (*PASS*/*WARN*/*FAIL*) into an overall per-driver rating using a linear severity score. A *FAIL* indicates a strong threshold exceedance and is therefore weighted twice a *WARN*, yielding a simple, interpretable metric that tolerates occasional minor deviations.

$$S \;=\; 2\,\#\text{FAIL} \;+\; \#\text{WARN},$$

$$\text{Rating}(S) = \begin{cases} \text{Good}, & S \le 2, \\ \text{Warning}, & 3 \le S \le 5, \\ \text{Bad}, & S \ge 6\ . \end{cases}$$

This 2:1 weighting reflects relative severity (a *FAIL* is a clearer breach of the threshold than a *WARN*) and preserves stability: small label fluctuations do not flip a driver from *Good* to *Bad*. The middle band (*Warning*) collects borderline cases for review.

Table 1: Per-driver severity summary (S = 2·#FAIL+#WARN).

Driver	#WARN	#FAIL	S	Rating
1	4	1	6	Bad
10	5	1	7	Bad
2	7	2	11	Bad
3	4	0	4	Warning
4	4	0	4	Warning
5	6	2	10	Bad
6	3	0	3	Warning
7	3	0	3	Warning
8	4	0	4	Warning

3.3 Machine Learning

3.3.1 *K-means clustering.*

Unsupervised clustering of K-means (k = 3) was applied per segment on standardized aggregated characteristics (acceleration / braking variability, coasting, use of cruise control, speed-related measures). Clusters were assigned semantic labels *Good*/*Warning*/*Bad* *post hoc* by ordering clusters by their mean fuel rate (`fuel_mean`): lowest ⟶ *Good*, middle ⟶ *Warning*, highest ⟶ *Bad*. We then examined cluster centroids (mean feature profiles) and visualised the result as per-segment heatmaps.

3.3.2 *SHAP with LightGBM model.*

As an orthogonal check of feature relevance, we applied SHAP to a separate LightGBM model predicting fuel rate; this diagnostic analysis is independent of clustering and highlights variables linked to higher consumption (Table 2).

4 Results

4.1 Statistical Thresholding Approach

Based on the analysis of related worke outlined in Section 2, we decided to benchmark driver efficiency based on selected driving features.We investigated two methods covering complementary metrics of acceleration and braking, namely:

- Percentile-based thresholds for gas pedal
- IQR method for brake pedal

Percentiles were chosen for the gas pedal because the signal is highly zero-inflated and not normally distributed, making distribution-aware thresholds more suitable. Braking behavior is irregular and heavy-tailed, where IQR offers a robust way to capture abnormal events. In essence, the IQR rule sets a dispersion-anchored cut-off above Q3-robust to heavy tails-whereas percentile thresholds fix the share of events flagged. Thresholds were determined by examining histograms of pedal deltas (Figure 2), ensuring that cutoffs meaningfully separated typical from extreme behavior. This procedure enabled transparent, segment-level benchmarking of driver performance.

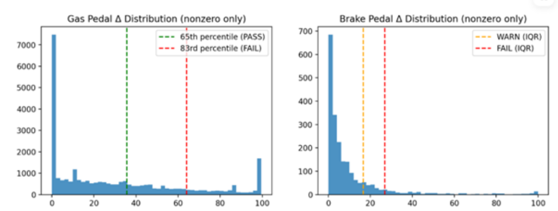


Figure 2: Histograms for both pedals

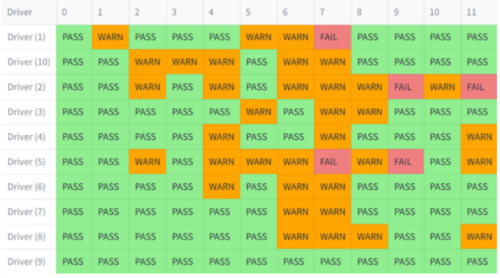
Threshold characterisation:

- Gas Pedal: We applied percentile-based thresholds (65th for WARN, 83rd for FAIL) to the gas pedal delta (change in 0,1 second). This approach better captures outlier acceleration behavior while avoiding over-penalizing normal operation. We removed windows where cruise control was active for more than 30% of the time to reduce automation bias in pedal measurements. It was chosen to balance isolating manual control with keeping enough observations.
- Brake Pedal: We applied an interquartile–range rule computed from the empirical distribution in each segment: with the third quartile $Q3$ and the interquartile range $\text{IQR} = Q3 - Q1$, we set *WARN* at $Q3 + 0.5\,\text{IQR}$ and *FAIL* at $Q3 + 1.5\,\text{IQR}$. It flags both frequent moderate excesses (*WARN*) and rare but severe braking events (*FAIL*) without over-penalising normal behaviour.

Certain segments in the driving scenario required strong braking due to test design (e.g., safety-critical stops). These were labelled as SAFETY and excluded from efficiency scoring, as they reflect controlled conditions rather than natural driving quality.

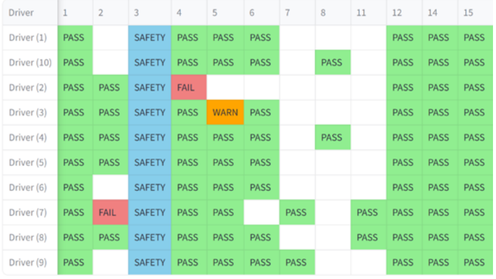
The resulting classifications are summarised as heatmaps (Figures 3 and 4), where rows correspond to drivers and columns to scenario segments. Cells are coloured green (PASS), orange

(WARN), and red (FAIL), providing an intuitive visual overview of performance variability. PASS/WARN/FAIL are segment-level, per-driver labels that state whether the segment was driven efficiently in terms of fuel use: PASS = efficient, WARN = borderline, FAIL = inefficient. These labels refer only to fuel consumption, not safety or travel time. Blank (white) cells indicate cases without an assigned label—either *SAFETY* segments excluded from scoring or driver–segment pairs with too few events to make a reliable decision.



Driver	0	1	2	3	4	5	6	7	8	9	10	11
Driver (1)	PASS	WARN	PASS	PASS	PASS	WARN	WARN	FAIL	PASS	PASS	PASS	PASS
Driver (10)	PASS	PASS	WARN	WARN	WARN	PASS	WARN	WARN	PASS	PASS	PASS	PASS
Driver (2)	PASS	PASS	WARN	PASS	WARN	PASS	WARN	WARN	WARN	FAIL	WARN	FAIL
Driver (3)	PASS	PASS	PASS	PASS	PASS	WARN	PASS	WARN	WARN	PASS	PASS	PASS
Driver (4)	PASS	PASS	PASS	PASS	WARN	PASS	PASS	WARN	PASS	PASS	PASS	WARN
Driver (5)	PASS	PASS	WARN	PASS	WARN	WARN	WARN	FAIL	WARN	FAIL	PASS	WARN
Driver (6)	PASS	PASS	PASS	PASS	WARN	PASS	WARN	WARN	PASS	PASS	PASS	PASS
Driver (7)	PASS	PASS	PASS	PASS	PASS	PASS	WARN	WARN	PASS	PASS	PASS	PASS
Driver (8)	PASS	PASS	PASS	PASS	PASS	PASS	WARN	WARN	WARN	PASS	PASS	WARN
Driver (9)	PASS	PASS	PASS	PASS	PASS	PASS	PASS	PASS	PASS	PASS	PASS	PASS

Figure 3: Heat map of the gas pedal through segments using percentiles method



Driver	1	2	3	4	5	6	7	8	11	12	14	15
Driver (1)	PASS		SAFETY	PASS	PASS	PASS				PASS	PASS	PASS
Driver (10)	PASS		SAFETY	PASS	PASS	PASS		PASS		PASS	PASS	PASS
Driver (2)	PASS	PASS	SAFETY	FAIL						PASS	PASS	PASS
Driver (3)	PASS	PASS	SAFETY	PASS	WARN	PASS				PASS	PASS	PASS
Driver (4)	PASS	PASS	SAFETY	PASS	PASS	PASS		PASS		PASS	PASS	PASS
Driver (5)	PASS	PASS	SAFETY	PASS	PASS	PASS				PASS	PASS	PASS
Driver (6)	PASS		SAFETY	PASS	PASS	PASS				PASS	PASS	PASS
Driver (7)	PASS	FAIL	SAFETY	PASS	PASS		PASS		PASS	PASS	PASS	PASS
Driver (8)	PASS	PASS	SAFETY	PASS	PASS	PASS			PASS	PASS	PASS	PASS
Driver (9)	PASS		SAFETY	PASS	PASS	PASS	PASS			PASS	PASS	PASS

Figure 4: Heat map of the brake pedal through segments using IQR method

4.2 Comparison of Thresholding and Clustering

A focused comparison was carried out on three representative track segments: Segment 1, Segment 8, and Segment 4 using the two complementary methods described in Section 3 (statistical thresholding and K-means clustering). For visualization only, we projected standardised features onto two principal components (PCA) per segment; clustering and label assignment were performed in the original standardised space.

4.2.1 Segment 1 (Steady Acceleration). The percentile method flagged only one driver as exceeding the 'FAIL' threshold, while most achieved the 'PASS' status. The clustering of K-means produced a tightly grouped 'Good' cluster for most drivers, with a single 'Bad' outlier (visible in PCA as an isolated point on the positive PC1 axis). Agreement between methods was high (>85 %), suggesting that, in simpler acceleration scenarios, single-feature metrics and multidimensional clustering agree well.

4.2.2 Segment 4 (Prolonged Uphill Driving). Here the disagreement was most pronounced. The percentile rule classified many drivers as *PASS* because their maximum throttle did not exceed the cut-off. In contrast, K-means frequently assigned them to *Warning* or *Bad*. The 2D PCA projection (Figure 6) shows these drivers displaced from the *Good* centroid, driven by sustained high-load throttle (elevated accelerator mean/variance), low coasting, and reduced cruise-control usage—patterns that the single-peak percentile metric does not penalize. This highlights clustering's sensitivity to cumulative demand and multi-feature context, whereas the percentile approach captures only isolated exceedances.

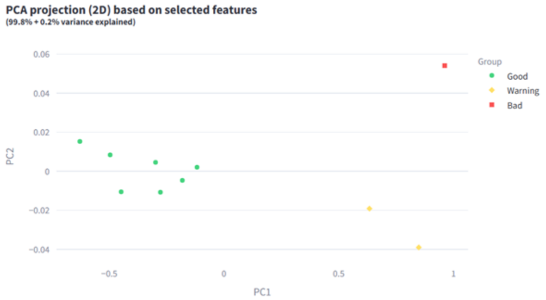


Figure 5: K-means graph for 1st segment

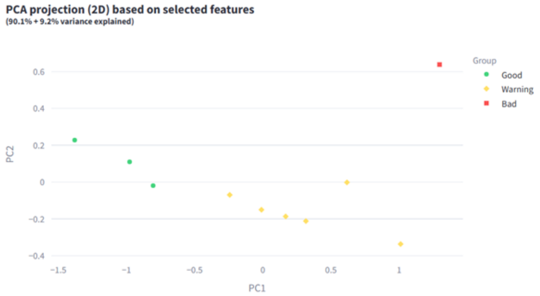


Figure 6: K-means graph for 4th segment

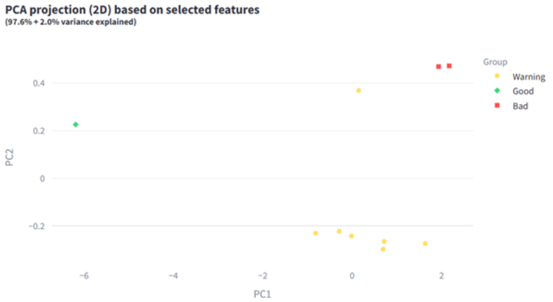


Figure 7: K-means graph for 8th segment

4.2.3 Segment 8 (Complex Curve–Acceleration Mix). This segment showed more divergence. The percentile method marked several drivers as 'WARN' due to short bursts of high throttle, while K-means placed some of these drivers in the 'Good' cluster. PCA visualization revealed that these drivers exhibited smoother braking and higher coasting ratios, which the clustering model positively weighted. This highlights a key difference: the statistical approach penalizes isolated peaks, whereas clustering balances them against compensatory behaviors.

4.2.4 Cross-approach Observations. The alignment was strongest in steady demand scenarios (Segment 1), weaker in mixed behavior contexts (Segment 8), and lowest in sustained load conditions (Segment 4). Statistical thresholding offers high interpretability and segment-level clarity, but may overlook multi-feature inefficiencies. K-means clustering captures complex, composite behavior and can sometimes reclassify drivers that the percentile method labels as efficient. It would be interesting for future work to implement more driver features and analyse in depth which have a different effect.

We additionally investigated the alignment between model-based feature importances and expert knowledge/domain expectations using SHAP.

Table 2: Top-5 features per class

Class	Top 1	Top 2	Top 3	Top 4	Top 5
Bad	AccelerationPedal	Speed	Acceleration	SteeringWheelAngle	BrakePedal
Medium	Speed	AccelerationPedal	Acceleration	SteeringWheelAngle	BrakePedal
Good	AccelerationPedal	Speed	Acceleration	SteeringWheelAngle	BrakePedal
Perfect	AccelerationPedal	Speed	Acceleration	SteeringWheelAngle	BrakePedal

Table 2 presents the five most influential features for each consumption class (*Bad*, *Medium*, *Good*, *Perfect*), ranked by their mean absolute SHAP value. The model consistently identifies *AccelerationPedal* and *BrakePedal* among the top-ranked features across multiple classes, in line with the statistical benchmark results from Section 4.1, where pedal usage was also the dominant indicator of inefficient driving events. This agreement confirms that the machine learning approach captures the same domain-relevant control inputs as the thresholds defined by the expert, while also highlighting secondary but relevant factors such as *Speed*, *Acceleration*, and *SteeringWheelAngle*.

4.3 Pareto Front of Time–Fuel Trade-Offs

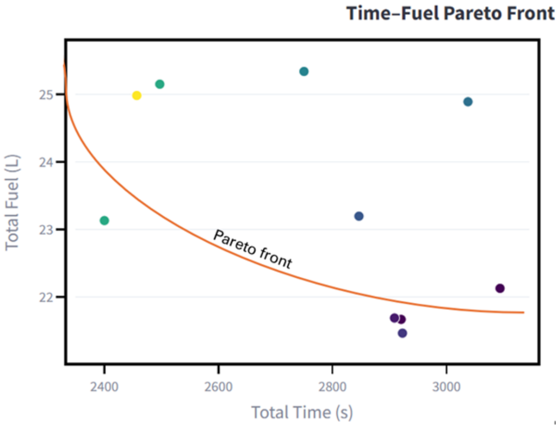


Figure 8: Pareto front

An interesting point of view would be to also consider the temporal information. Fuel consumption may reduce costs, but time is also quite important. Figure 8 plots the total time against the total fuel per driver. A driver is Pareto efficient if no other driver is faster and uses less fuel; these form the lower-left frontier. The points to the upper-right are dominated and can improve at least one objective without worsening the other. We obtain the frontier by non-dominated sorting of per-driver $(Time, Fuel)$ totals and colour points by their K-means group, explicitly linking global efficiency to the segment-level patterns identified earlier.

5 Discussion

This comparative study shows that rule-based thresholding remains highly interpretable and aligns with prior work, while K-means clustering reveals multi-feature patterns that affect efficiency. In practice, percentile rules flag isolated exceedances, whereas clustering captures cumulative demand and co-variation, explaining the discrepancies observed in segments such as Figure 6. Together, the methods are complementary: thresholding offers transparent guardrails; clustering provides a broader, context-aware view.

6 Conclusions

The results suggest that integrating both statistical and machine learning perspectives offers a more robust and nuanced driver assessment for fuel efficiency. While classical thresholding offers transparency, machine learning enables the discovery of complex patterns. Future work should further validate these findings to develop hybrid driver feedback systems. We only used SHAP diagnostically; a more systematic SHAP analysis would be interesting across models, segments, and time, to stabilize attributions and translate them into actionable feedback.

Acknowledgements

We thank the AAER Research d.o.o. team, led by CEO Matej Vengust, for access to simulator data and expert support. We also acknowledge support from the EDIH DIGI–SI project.

References

[1] Oscar Delgado, Felipe Rodríguez, and Rachel Muncrief. 2017. Fuel Efficiency Technology in European Heavy-Duty Vehicles: Baseline and Potential for the 2020–2030 Timeframe. White Paper. The International Council on Clean Transportation (ICCT), (July 2017). https://theicct.org/publication/fuel-efficiency-technology-in-european-heavy-duty-vehicles-baseline-and-potential-for-the-2020-2030-timeframe/.

[2] Hung Nguyen, George Tsaramirsis, Ilir Mborja, Dhimitraq Dervishi, Eriona Hoxha, Stavros Shiaeles, Anastasios Kavoukis, and Stamatios Vologiannidis. 2023. A data-driven framework for incentivising fuel efficient driving behaviour in heavy-duty vehicles. *J. Clean. Prod.*, 420, 139942. DOI: 10.1016/j.jclepro.2023.139942.

[3] Shuyan Chen, Hongru Liu, Yongfeng Ma, Fengxiang Qiao, Qianqian Pang, Ziyu Zhang, and Zhuopeng Xie. 2024. High fuel consumption driving behavior identification and causal analysis based on lightgbm and shap. *Res. Sq.* Preprint. DOI: 10.21203/rs.3.rs-4010652/v1.

[4] Alexander Meschtscherjakov, David Wilfinger, Thomas Scherndl, and Manfred Tscheligi. 2009. Acceptance of future persuasive in-car interfaces towards a more economic driving behaviour. In *AutomotiveUI 2009*. (Sept. 2009), 81–88. DOI: 10.1145/1620509.1620526.

[5] Charles Sullivan and Mark Franklin. 2010. An extended driving simulator used to motivate analysis of automobile fuel economy. In *Session 1: Tools, techniques, and best practies of engineering education for the digital generation.* (May 2010). DOI: 10.18260/1-2-1153-53783.

[6] Mathieu Maisonneuve. 2013. *Characterization of drivers' energetic efficiency: Identification and evaluation of driving parameters related to energy efficiency.* Master's thesis. Chalmers University of Technology. https://hdl.handle.net/20.500.12380/185531.

[7] Xiaohua Zhao, Yiping Wu, Jian Rong, and Yunlong Zhang. 2015. Development of a driving simulator based eco-driving support system. *Transportation Research Part C: Emerging Technologies*, 58, 631–641. Technologies to support green driving. DOI: https://doi.org/10.1016/j.trc.2015.03.030.

[8] Zhipeng Ma, Bo Nørregaard Jørgensen, and Zheng Ma. 2024. A scoping review of energy-efficient driving behaviors and applied state-of-the-art ai methods. *Energies*, 17, 2. DOI: 10.3390/en17020500.

[9] A McGordon, J E W Poxon, C Cheng, R P Jones, and P A Jennings. 2011. Development of a driver model to study the effects of real-world driver behaviour on the fuel consumption. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 225, 11, 1518–1530. DOI: 10.1177/0954407011409116.

[10] Thomas J. Daun, Daniel G. Braun, Christopher Frank, Stephan Haug, and Markus Lienkamp. 2013. Evaluation of driving behavior and the efficacy of a predictive eco-driving assistance system for heavy commercial vehicles in a driving simulator experiment. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, 2379–2386. DOI: 10.1109/ITSC.2013.6728583.

Automated Explainable Schizophrenia Assessment from Verbal-Fluency Audio

Rok Rajher
Jure Žabkar
rr3244@student.uni-lj.si
jure.zabkar@fri.uni-lj.si
University of Ljubljana,
Faculty of Computer and Information Science,
Ljubljana, Slovenia

Abstract

Schizophrenia is associated with cognitive impairments that are difficult to assess with traditional neuro-psychological tests. Currently, these tests are manually administered by clinical doctors and rely on subjective assessment of patient's behavior, self-reported symptoms, medical history, and mental state. Recent advances in deep learning substantially improved automatic speech recognition (ASR), and large language models (LLMs), enabling the development of computational tools that can partially automate aspects of psychiatric assessment. We present the first fully automated classification of individuals with schizophrenia based on verbal-fluency tests conducted in Slovene language. Our multi-stage pipeline involves audio preprocessing, automatic transcription using the Truebar ASR model, the extraction of meaningful verbal and non-verbal features, and learning a machine learning model. The Explainable Boosting Machine (EBM) trained on the obtained feature set achieved the best overall performance.

Keywords

schizophrenia, automatic speech recognition, large language models, verbal-fluency tasks, machine learning

1 Introduction

Schizophrenia is a chronic and severe mental disorder [8, 11] that affects how a person thinks, feels, and behaves. As a psychotic disorder it is characterized by a combination of disorganized thinking and behavior, hallucinations, and delusions [2, 14]. The symptoms have major implications on individual's social life and can lead to a lifelong care [1, 7]. Schizophrenia affects about 1% of the population worldwide [9].

Currently, there is no objective or standardized diagnostic test for schizophrenia. The most widely used diagnostic frameworks in clinical practice are the DSM-5 [2] and the ICD-11 [14]. With rapid improvements in automatic speech recognition (ASR), large language models (LLMs), and machine learning, there is rising interest in computational tools that support, augment, or partially automate aspects of psychiatric assessment.

Clinicians have long noted that schizophrenia systematically affects speech in two ways:

(1) how people speak: acoustic-prosodic markers such as pause structure, speech rate, and prosodic variability, and
(2) what they say: lexical-semantic markers such as category switching, perseverations, and vocabulary diversity.

These are best observed during verbal-fluency tasks - short, standardized, low-burden, and already used in clinical practice. Our main hypothesis is that short recordings of Slovene verbal-fluency tasks contain sufficient discriminative signal, captured by acoustic and semantic features, to separate individuals with schizophrenia from healthy controls.

In this paper, we present automated machine learning pipeline for the detection and explanation of schizophrenia, leveraging the capabilities of ASR models and state-of-the-art LLMs. The tests were conducted in the Slovene language and consisted of two one-minute subtasks: (1) a semantic fluency task, where participants were asked to list as many animal names as possible, and (2) a phonetic fluency task, where participants were instructed to generate words beginning with the letter 'L'. The approach is based on audio recordings of verbal fluency tests collected by Marinković [10]. Our results can be directly compared to those reported by Marinković [10], where the transcription and analysis of the tests were performed manually. The details of our study are extensively described in [13].

2 Methods

2.1 Participants

The dataset comprises of 126 participants: 58 individuals with a clinical diagnosis of schizophrenia (SH), and 68 healthy controls (HC). All individuals in the SH group were patients admitted to the University Psychiatric Clinic Ljubljana. All participants were adults aged 18 years or older and gave consent to being part of the experiment.

Standard demographic information was collected for each participant, including age, gender, highest level of education, academic performance (school grades), marital status, and employment status. The dataset is balanced with respect to age and gender.

For participants diagnosed with schizophrenia, additional clinical information was recorded: illness duration, number of hospitalizations, and the presence of chronic or co-occurring health conditions. The median illness duration among individuals with schizophrenia was 10 years, with a median of 4 hospitalizations.

The study was approved by the Medical Ethics Committee of the Republic of Slovenia (approval number: 0120-51/2024-2711-4). All participants received a detailed explanation of the study procedures and provided written informed consent prior to participation.

Information Society 2025, Ljubljana, Slovenia
© 2025 Copyright held by the owner/author(s).
https://doi.org/10.70314/is.2025.skui.2350

Measure	SH	HC
Total Participants	58	68
Male Distribution	29	35
Female Distribution	29	33
Median Age (years)	45	46.5
Median Primary School Grade	3	5
Median High School Grade	3	4
Median Illness Duration (years)	10	–
Median Number of Hospitalizations	4	–
Prevalent Education Level	Elem.	HS
Prevalent Marital Status	Married	Married
Prevalent Employment Status	Retired	Employed

Table 1: Demographic and clinical characteristics of the participants.

2.2 Testing procedure

Each participant completed a verbal fluency test consisting of two sub-tasks:

(1) **Phonetic fluency task:** participants were asked to produce as many Slovene words as possible beginning with the letter 'L'. Proper nouns, including names of people or places, were not allowed. The task lasted 62 seconds in total: during the first 2 seconds, the letter 'L' was displayed on the screen, followed by 60 seconds for verbal response.
(2) **Semantic fluency task:** participants were instructed to name as many animals as possible in the Slovene language. Pet names and proper nouns were not allowed. The task duration was 60 seconds.

The testing procedure was standardized: each individual was seated in front of a laptop computer. After reading the instructions for the phonetic fluency task, the participant pressed a key to begin, initiating the countdown. After completing the first task, the instructions for the second task (semantic fluency) were displayed. Again, the participant initiated the task by pressing a key when ready. This concluded the verbal fluency test.

Healthy participants were tested at the Faculty of Computer and Information Science, University of Ljubljana, while individuals with schizophrenia were assessed at the University Psychiatric Clinic Ljubljana. To ensure consistency across conditions, all recordings were conducted in quiet, isolated rooms to eliminate possible noise and distractions.

All WAV files then underwent the same audio enhancement pipeline: (i) dynamic range compression to reduce variability due to speaking loudness and microphone distance, and (ii) loudness normalization to achieve consistent perceived loudness across recordings. These steps were implemented with standard functions from pydub and applied identically to both sites prior to feature extraction.

2.3 Data Format

The final dataset consists of 126 WAV audio recordings, one per participant, captured using the built-in laptop microphone during the test sessions. The audio tracks are encoded in uncompressed PCM format at a sampling rate of 44.1 kHz with a single (mono) audio channel. Additionally, there are 126 corresponding CSV files containing timestamps that indicate the start and end times of each subtask. Together, these audio and timestamp files serve as the primary data sources for all subsequent audio- and speech-based analyses.

3 Preprocessing

3.1 Audio Data Preparation

The WAV recordings were initially divided into two distinct audio segments using the provided timestamp files: (1) a segment corresponding to the phonetic verbal fluency task and (2) a segment corresponding to the semantic verbal fluency task.

Both audio segments were then processed through a series of audio enhancement steps:

(1) **Dynamic range compression:** To improve audio quality and ensure uniformity, downward dynamic range compression (threshold = -20.0 dBFS, ratio = 4:1, attack time = 5 ms, release time = 50 ms) was applied to each segment. This reduces the volume gap between the quietest and loudest parts of a signal [6].
(2) **Loudness normalization:** adjusting each segment to a target level of -20 dBFS. This ensured consistent perceived loudness across all recordings, reducing variability from differences in speaker volume, room acoustics, or microphone distance.
(3) **Final output:** Finally, the two fully processed segments per participant (phonetic and semantic) were saved as separate WAV files. These files constitute the final audio data used for all subsequent analyses.

All of the described steps were implemented using standard functions provided by the pydub library.

3.2 Feature Engineering

After automated transcriptions have been processed we performed feature engineering. Based on clinical knowledge, we created meaningful features that serve as reliable markers for distinguishing between individuals with and without schizophrenia. Three core symptoms of schizophrenia are directly applicable to our verbal-fluency tasks: disorganized speech, disorganized behavior, and negative symptoms. The primary rationale behind our feature construction is grounded in these core symptom domains.

Audio recordings are represented in two forms: (1) as text, derived from automated ASR transcriptions, and (2) as spectrograms – a visual representation of the frequency content of the audio signal over time. We constructed two groups of features:

(1) **Verbal features:** 39 features derived from the automated text transcriptions. These features aim to quantify disorganized speech, e.g. number of phrases produced per second.
(2) **Non-verbal features:** 17 features extracted directly from the spectrograms of the audio recordings, these features target prosodic elements such as pitch and vocal control, which are key indicators of negative symptoms like blunted affect and disorganized behavior; e.g. Mean pitch, representing the speaker's average vocal pitch.

3.3 Automated Transcription

The most critical step in the preprocessing of audio recordings is the generation of automated transcriptions. These ASR-derived transcriptions serve as the primary input for nearly all subsequent stages of feature extraction and machine learning analysis. We employed the ASR model Truebar 24.05, a state-of-the-art speech recognition system for the Slovene language. The model was developed by the company Vitatis in collaboration with the Laboratory for Data Technologies at the Faculty of Computer and Information Science. Using Truebar API we programmatically uploaded each audio file and in response receive the corresponding transcribed words along with their start and end timestamps.

3.4 Transcription Adjustment

The output of the ASR system consists of transcribed words along with their associated timestamps. These transcriptions may include irrelevant content such as filler words. We used the DSPy library—a Python framework that enables declarative programming for prompting LLMs in a modular and programmatic way in combination with GPT-4o model. The transcription adjustment process was divided into two sequential steps:

(1) **Transcription filtering:** The raw transcription output from the Truebar ASR model was first passed to the GPT-4o model along with a description of the verbal fluency task and its rules. The model was instructed to retain only the words it considered to be relevant without modifying the words themselves.

(2) **Transcription correction:** The filtered transcription was then forwarded to the model in a second pass. With the same task context and rules provided, the model was now asked to adjust incorrectly transcribed words to what it inferred the participant likely intended to say. A word could potentially also be a neologism, we explicitly instructed the model to apply corrections only when the intended word was judged to be clear and obvious; otherwise, the word was left unchanged. For example, a misrecognized word like 'lon' would be corrected to 'slon' (elephant), whereas unclear or ambiguous cases were preserved as-is.

3.5 Adding Semantic Meaning

After filtering and correcting the transcriptions, we tagged each word with semantic annotations relevant to the verbal fluency task. These semantic features are crucial for distinguishing between HC and SH, as they capture subtle linguistic anomalies commonly associated with schizophrenia. We used DSPy framework in combination with the GPT-4o language model to perform automated semantic tagging. The model was provided with task-specific instructions and context for each word. For each transcribed word, we extracted the following semantic tags:

- **Intrusion:** The word is semantically unrelated to the target category (e.g. non-animal word during the animal naming task). Intrusions are often more frequent in individuals with schizophrenia and reflect impaired cognitive control and semantic memory organization [5].
- **Stiltedness:** Marks whether the word appears overly formal, unusual, or unnatural in everyday speech. Stilted language is a known linguistic feature in schizophrenia and may signal underlying disruptions in pragmatic language use [12].
- **Neologism:** a newly coined or nonsensical word not found in the lexicon. Neologisms are characteristic of disorganized thought and speech, and are especially relevant in schizophrenia research [3].
- **Word description (semantic task only):** A general, page-long descriptive summary of the word. For animals, this includes common features such as appearance, habitat, and behavior—providing a semantic embedding that captures how the word is typically perceived by the general population. In the case of neologisms, the semantic meaning was still applied based on what the word could plausibly represent or mean, allowing the model to assign an approximate semantic embedding even for novel or invented terms. This feature is used only for the semantic task, where meaning-based associations between words are essential.

3.6 Data Analysis

We trained and evaluated several machine learning models using these features. To ensure robust evaluation, we applied stratified 10-fold cross-validation. Performance was assessed using accuracy, sensitivity, specificity, and the area under the receiver operating characteristic curve (AUC). The *Explainable Boosting Machine (EBM)* consistently achieved the best results when trained on the full feature set. We additionally examined the top 10 most informative features to assess model interpretability. This approach enables us to understand better which et deficits are most prominent in individuals with schizophrenia and may be useful for targeted clinical interventions.

4 Results

We observe that the obtained ML models perform similarly when using the verbal (V) and non-verbal (N) feature sets separately, achieving an average AUC of 0.83 on both datasets. In the combined feature set (VN), the average performance improves across all metrics: AUC 0.86, CA 0.76, Sens. 0.69, Spec. 0.82, PPV 0.76, and F1 0.73. The EBM trained on the combined feature set (VN) achieved the best overall performance: AUC 0.90, CA 0.82, Sens. 0.76, Spec. 0.87, PPV 0.83, and F1 0.79.

To probe whether education could drive the observed performance, we examined models trained on verbal (V) and non-verbal (N) features separately, in addition to the combined set (VN). Verbal features are more likely to reflect educational attainment (e.g., lexical diversity, category switching), whereas core acoustic markers (e.g., pause structure, longest silent pause) are less dependent on education [4]. In our 10-fold CV, V and N models performed comparably, and VN performed best. This suggests that education alone is unlikely to explain the classification.

4.1 Global interpretation

The overall feature importance (FI) across the entire dataset is used for global interpretation of the model. We calculate it as the average absolute contribution of each feature across all samples:

$$\mathrm{FI}_j = \frac{1}{n} \sum_{i=1}^{n} \left| f_j(x_{i,j}) \right|, \tag{1}$$

where n is the total number of samples, and $f_j(x_{i,j})$ is the contribution of feature j for instance i. FI measures how strongly each feature influences the model's predictions on average.

Globally most important features are: (1) comb_pho_lev2_avg - the Levenshtein similarity between the filtered and adjusted transcriptions, which indicates impaired speech fluency, (2) animal_tempo_max_gap_percent - captures the longest silent pause during the semantic task, (3) animal_sem_cont_max_coherence_index, animal_sem_cont_kurt_coherence_index, and ltest_sem_stat_min_coherence_index - the first two capture the word-to-word coherence, while the third captures the lowest phonetic similarity between consecutive words during the phonetic task, (4) `comb_osmile_F0From27.5Hz_stddeNorm_avg` - the standard deviation of pitch; highlights variability in vocal pitch — a marker of prosodic irregularity often observed in individuals with schizophrenia.

4.2 Local interpretation

Each individual prediction can be explained through the positive/negative contribution of each feature. Features with positive contributions increase the log-odds in favor of the schizophrenia class, while features with negative contributions decrease the log-odds, shifting the prediction toward the healthy control class. An example for a severe schizophrenia case is shown in Fig. 1

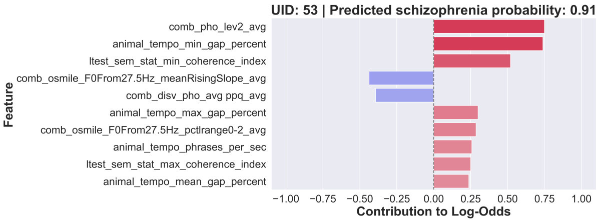


Figure 1: Local feature importance plot for a severe schizophrenia case as predicted by the EBM model. Red bars indicate contributions toward the schizophrenia class, and blue bars toward the healthy control class.

The corresponding textual explanation was generated by `GPT-4o` model: The results from the verbal fluency test indicate several features often associated with schizophrenia. Short pauses between utterances may suggest rushed or pressured speech, which can be a sign of reduced speech planning. Low semantic coherence in structured tasks may indicate the intrusion of unrelated thoughts or semantic derailment. Additionally, long pauses between utterances can reflect cognitive slowing or difficulty with word retrieval. These features collectively suggest the possibility of schizophrenia. The results suggest that, on average, the models are able to rank individuals effectively (high AUC); they can distinguish between HC and SH in terms of relative probability. The low CA, sensitivity, PPV, and F1 scores suggest that the chosen classification threshold of 0.5 may not be optimal. This issue was further addressed by evaluating the ROC curve of the best-performing model to explore whether an alternative classification threshold could improve the identification of positive cases; we observed that both the Youden-optimal threshold and the F1-optimal threshold are approximately 0.49, which differs negligibly from the used value of 0.5.

The performance of our best model, EBM, shows its strong ranking ability, and balanced classification performance on both classes.

Limitations and Future Work

Although our dataset is well-balanced, the sample size (126) is rather small; additional samples would improve model generalizability and robustness. Audio quality could be improved by using professional microphones instead of built-in laptop microphones, which would enhance transcription accuracy. Due to obtaining the audio recordings at two locations, a residual site effect cannot be fully excluded. We mitigated the risk by (i) using identical task instructions and timing in quiet rooms at both sites, (ii) applying uniform dynamic range compression and loudness normalization to all audio, and (iii) demonstrating that transcript-only models (verbal features) remain predictive, indicating that performance is not driven by background acoustics. Future studies should also include participants with other psychiatric conditions, such as major depressive disorder or bipolar disorder.

Conclusion

We developed and evaluated an automated, explainable pipeline for schizophrenia assessment using 126 verbal-fluency audio recordings (healthy controls: 68; schizophrenia: 58). The pipeline comprises audio pre-processing, automatic transcription with the Truebar ASR model, and extraction of verbal (transcript-derived) and non-verbal (acoustic/temporal) features. The features were then used in training and evaluation of several classical machine-learning models.

Across models, combining verbal and non-verbal features consistently yielded the strongest results. The Explainable Boosting Machine achieved the highest performance: CA 0.82, Sens. 0.76, Spec. 0.87, PPV 0.83, F1 0.79, and AUC 0.90. Due to the EBM's inherent interpretability, we produced global explanations and local explanations (per-instance contribution plots), complemented by GPT-4o–generated textual summaries. A high model performance and associated explanations provide a firm ground for potential decision support system in clinical practice.

5 Acknowledgments

This work was partially supported by the Slovenian Research Agency (ARIS), research program Artificial Intelligence and Intelligent Systems (Grant No. P2-0209).

References

[1] Bandar AlAqeel and Howard C. Margolese. 2012. Remission in schizophrenia: Critical and systematic review. *Harvard Review of Psychiatry* 20, 6 (2012), 281–297.

[2] American Psychiatric Association. 2013. *Diagnostic and statistical manual of mental disorders (5th ed.).* American Psychiatric Publishing, Arlington, VA.

[3] Janna N. De Boer, Sanne G. Brederoo, Alban E. Voppel, and Iris E. C. Sommer. 2020. Anomalies in language as a biomarker for schizophrenia. *Current Opinion in Psychiatry* 33, 3 (2020), 212–218.

[4] J. N. De Boer, A. E. Voppel, S. G. Brederoo, H. G. Schnack, K. P. Truong, F. N. K. Wijnen, and I. E. C. Sommer. 2023. Acoustic speech markers for schizophrenia-spectrum disorders: a diagnostic and symptom-recognition tool. *Psychological Medicine* 53, 4 (March 2023), 1302–1312.

[5] Flavia Galaverna, Adrián M. Bueno, Carlos A. Morra, María Roca, and Teresa Torralva. 2016. Analysis of errors in verbal fluency tasks in patients with chronic schizophrenia. *The European Journal of Psychiatry* 30, 4 (2016), 305–320.

[6] Dimitrios Giannoulis, Michael Massberg, and Joshua D. Reiss. 2012. Digital dynamic range compressor design—A tutorial and analysis. *Journal of the Audio Engineering Society* 60, 6 (2012), 399–408.

[7] Josep Maria Haro, Diego Novick, Jordan Bertsch, Jamie Karagianis, Martin Dossenbach, and Peter B. Jones. 2011. Cross-national clinical and functional remission rates: Worldwide Schizophrenia Outpatient Health Outcomes (W-SOHO) study. *The British Journal of Psychiatry* 199, 3 (2011), 194–201.

[8] Thomas R. Insel. 2010. Rethinking schizophrenia. *Nature* 468, 7321 (2010), 187–193.

[9] Stephen R. Marder and Tyrone D. Cannon. 2019. Schizophrenia. *The New England journal of medicine* 381, 18 (2019), 1753–1761. doi:10.1056/NEJMra1808803

[10] Mila Marinković. 2024. Analysis of speech fluency in patients with schizophrenia [Master's Thesis, University of Ljubljana, Faculty of Computer and Information Science].

[11] Robert A. McCutcheon, Tiago Reis Marques, and Oliver D. Howes. 2020. Schizophrenia—An overview. *JAMA Psychiatry* 77, 2 (2020), 201–210.

[12] Victor Peralta, Manuel J. Cuesta, and Jose de Leon. 1992. Formal thought disorder in schizophrenia: A factor analytic study. *Comprehensive Psychiatry* 33, 2 (1992), 105–110.

[13] Rok Rajher. 2025. Automatic Generation of Explanations in Diagnosing Schizophrenia Using Speech Fluency Testing [Master's Thesis, University of Ljubljana, Faculty of Computer and Information Science].

[14] World Health Organization. 2022. ICD-11: 6A20 Schizophrenia. Retrieved from https://icd.who.int/browse/2025-01/mms/en#1683919430.

Mapping Medical Procedure Codes Using Language Models

Mariša Ratajec
University of Ljubljana, Faculty of Electrical Engineering; Jožef Stefan Institute
Ljubljana, Slovenia
ratajec.marisa@gmail.com

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

Nina Reščič*
Jožef Stefan Institute
Ljubljana, Slovenia
nina.rescic@ijs.si

Abstract

Aligning medical procedure codes across national classification systems is a challenging task. We investigate the mapping of Slovenian KTDP expressions to German OPS codes using fuzzy matching, biomedical language models (BioBERT, GatorTron), a hybrid approach, and ChatGPT. In the absence of ground truth, we assess consistency across methods and conduct manual reviews. Results show that differences in code structure and expression detail pose major barriers to alignment. Expert validation will be essential for improving accuracy.

Keywords

procedure coding, KTDP, OPS, semantic similarity, BioBERT, fuzzy matching, GatorTron, ChatGPT

1 Introduction

Different countries maintain their own national classification systems for medical procedures, used for clinical documentation, reimbursement, public reporting, and statistical analysis. In Slovenia, healthcare professionals rely on a domestic procedural coding system, while in Germany, the Operationen- und Prozedurenschlüssel (OPS) is used.

At the University Medical Centre (UMC) Ljubljana in Slovenia, interest has emerged in matching expressions from the Klasifikacija terapevtskih in diagnostičnih postopkov in posegov (KTDP) with the German OPS classification system. The purpose is to allow international reporting, cost estimation, and comparative analysis of healthcare procedures.

1.1 Problem Outline

Aligning Slovenian procedural expressions with German OPS codes is a complex task. The Slovenian dataset contains approximately 6,000 expressions, whereas the German OPS classification includes more than 60,000 distinct entries, covering multiple levels of specificity in various medical domains. Manual mapping is time-consuming and impractical, primarily due to the size of datasets and the absence of convenient tools for efficient code retrieval and comparison.

To address this challenge, we explored the development of computational approaches to support and accelerate the mapping process. Due to the nature of the data and the semantic variation between codes, we tested several techniques, including fuzzy string matching, semantic similarity scoring, and large language models (LLMs), such as BioBERT, GatorTron, and OpenAI models. We also explored a hybrid approach that integrates fuzzy matching with LLM-derived semantic embeddings.

In this paper, we present the application of the selected methods for aligning Slovenian KTDP procedure expressions with German OPS codes. We evaluate their performance, limitations and discuss key challenges associated with this type of code matching problem.

2 Methodology

2.1 Datasets

2.1.1 Slovenian Dataset. The Slovenian dataset is based on the Klasifikacija terapevtskih in diagnostičnih postopkov in posegov (KTDP)[6], version 11, which has been officially implemented nationwide since 1 January 2023. This classification system is used to code medical procedures in all levels of healthcare in Slovenia and is structurally aligned with the Australian Classification of Health Interventions (ACHI), adapted to the local context.

KTDP consists of 20 chapters, each covering a different clinical domain. The chapters are organised primarily by body system (e.g. nervous, endocrine, cardiovascular), with additional sections dedicated to dental care, imaging services, radiation oncology, and interventions not elsewhere classified. Each chapter is subdivided into multiple blocks, which group related procedures under shared headings.

In total, the classification includes approximately 6,000 unique procedures. Each is assigned a specific code in a structured numeric format composed of a five-digit base and a two-digit extension (e.g. 36564-00).

2.1.2 German Dataset. The German dataset is based on Operationen und Prozedurenschlüssel (OPS), version 2024 [2], which is officially used nationwide for coding medical procedures. Maintained by the Federal Institute for Drugs and Medical Devices (BfArM), OPS is revised annually. It is derived from the WHO's International Classification of Procedures in Medicine (ICPM) and adapted to the German healthcare system.

The classification is organised into six main chapters, covering the following clinical domains: diagnostic measures (Chapter 1), imaging diagnostics (Chapter 3), surgical procedures (Chapter 5), medications (Chapter 6), non-operative therapeutic measures (Chapter 8), and supplementary measures (Chapter 9). Each chapter is further subdivided into categories and blocks, which group related procedures based on functional or anatomical criteria.

OPS comprises approximately 60,000 unique procedures. Each is assigned a hierarchical alphanumeric code, consisting of a four-digit base and optional numeric or alphanumeric extensions (e.g. 5-384.50 or 8-844.5c). The coding system follows a structured hierarchy, beginning with the chapter number (e.g. 5 for surgical procedures), followed by a category (e.g. 5-38 for vascular

[*] Corresponding author

Information Society 2025, Ljubljana, Slovenia

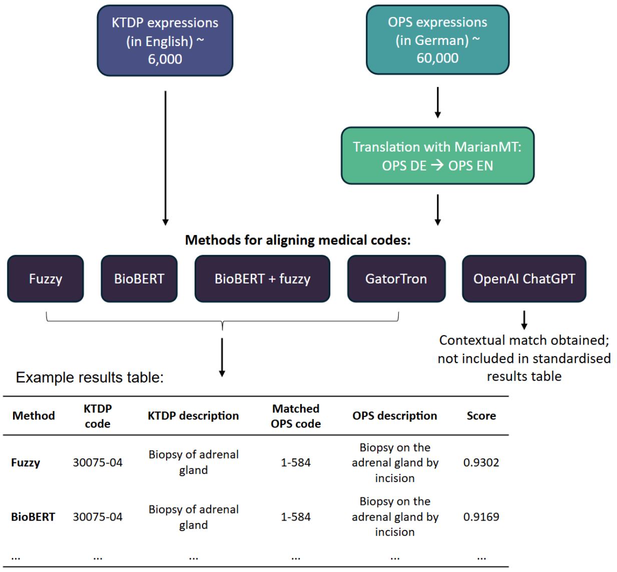
https://doi.org/10.70314/is.2025.skui.9648

surgery) and subcategories (e.g. 5-384 for specific surgical techniques). The digits and characters after the dot denote the exact intervention.

2.1.3 Differences and Similarities between Datasets. Although both classification systems serve a similar purpose, they differ in structure and level of detail. The German dataset includes very specific and thoroughly described procedures, clearly outlining each individual service. The Slovenian system, on the other hand, uses broader and more general descriptions, without the same amount of detail or length.

Moreover, there is limited direct lexical overlap between the two datasets. Even when procedures are conceptually similar, their descriptions often differ in phrasing, level of specificity, or use of synonyms. As a result, one-to-one matching is rarely straightforward and requires both structural alignment and semantic interpretation.

2.2 Pipeline


KTDP expressions (in English) ~ 6,000
OPS expressions (in German) ~ 60,000
Translation with MarianMT: OPS DE → OPS EN
Methods for aligning medical codes:
Fuzzy
BioBERT
BioBERT + fuzzy
GatorTron
OpenAI ChatGPT
Contextual match obtained; not included in standardised results table
Example results table:

Method	KTDP code	KTDP description	Matched OPS code	OPS description	Score
Fuzzy	30075-04	Biopsy of adrenal gland	1-584	Biopsy on the adrenal gland by incision	0.9302
BioBERT	30075-04	Biopsy of adrenal gland	1-584	Biopsy on the adrenal gland by incision	0.9169
...	...	...	...	...	...

Figure 1: Overview of the matching pipeline and example results. KTDP expressions in English were aligned to translated OPS expressions using five methods: fuzzy matching, BioBERT, a combined BioBERT+fuzzy approach, GatorTron, and OpenAI ChatGPT. All methods except ChatGPT produced structured outputs with match scores, as shown in the example results table. ChatGPT returned only contextual matches without comparable scoring and was therefore excluded from the standardised evaluation table.

The overall process is summarised in a pipeline diagram (Figure 1), which outlines each step — from dataset preparation and translation to the application of matching methods and the structure of resulting outputs. Each component of the pipeline is described in detail in the following subsections.

2.2.1 Translation. Since Slovenian KTDP expressions were already available in English, the German OPS procedure names were translated to English to enable semantic comparison. For this purpose, we used the MarianMT model (`Helsinki-NLP/opus-mt-de-en`) [4], a transformer-based neural machine translation model. Although not specifically fine-tuned for clinical texts, MarianMT has demonstrated strong performance in medical translation tasks, particularly for structured terminology [5], making it a suitable and practical choice for this application.

2.2.2 Language-based code pairing. To perform code matching, we initially applied a language-based code pairing approach using fuzzy matching, implemented via the RapidFuzz library [1]. Fuzzy matching is particularly useful in cases where expressions differ slightly in wording, structure, or spelling. We applied the token set ratio, which compares the sets of unique words in two strings and calculates a similarity score based on the overlap of unique tokens, with edit distance applied to the remaining unmatched parts. This method is insensitive to word order and robust to minor variations in phrasing. Using this approach, each English KTDP expression was compared with all translated OPS descriptions. For each KTDP entry, we selected the best matching OPS procedure based on the highest fuzzy similarity score and recorded the corresponding code, description, and score for further analysis.

2.2.3 Semantic-based code pairing. As a second approach, we applied a semantic-based code pairing approach using contextual embeddings derived from transformer-based language models. Specifically, we tested two pretrained models: `pritamdeka/BioBERT-mnli-snli-scinli-scitail-mednli-stsb` [3], a SentenceTransformer variant of BioBERT fine-tuned on biomedical and inference tasks, and `UFNLP/gatortron-base` [10], a GatorTron model pre-trained on large-scale clinical corpora. Both models were selected for their strong performance in biomedical language understanding [7] and to investigate how model choice influences the quality of semantic code alignment.

Using each model, both KTDP expressions and translated OPS descriptions were encoded into dense semantic vectors. Cosine similarity was then computed between each KTDP embedding and all OPS embeddings to assess semantic closeness. As in the previous approach, the top matching OPS procedure for each KTDP expression was selected and recorded following the same procedure as before.

2.2.4 Combined code pairing. In addition to the individual use of semantic and lexical methods, we implemented a hybrid matching approach that combines the strengths of both. Specifically, we integrated semantic similarity scores obtained from BioBERT embeddings with lexical similarity scores derived from fuzzy matching (token set ratio). For each KTDP expression, both similarity measures were computed independently against all translated OPS descriptions. The final similarity score for each pair was calculated as a weighted average:

$$\text{score}_{\text{final}} = w_{\text{semantic}} \cdot \text{score}_{\text{semantic}} + w_{\text{lexical}} \cdot \text{score}_{\text{lexical}}$$

We experimented with two weighting schemes: one with equal weights ($w_{\text{semantic}} = 0.5$, $w_{\text{lexical}} = 0.5$) and another prioritising semantic similarity ($w_{\text{semantic}} = 0.7$, $w_{\text{lexical}} = 0.3$), to assess how different balances influence match quality. For each KTDP expression, the OPS description with the highest combined score was selected and recorded along with the corresponding code and similarity score.

This approach was motivated by practical observations in the literature, where combining surface-level and context-aware similarity often yields more robust results, especially in cases where purely semantic models overlook minor wording differences or where lexical methods fail to capture deeper conceptual alignment [9].

2.2.5 ChatGPT code pairing. As a final exploratory method, we used a custom ChatGPT instance (GPT-4o, OpenAI) [8] to evaluate the potential of conversational large language models (LLMs) for code matching. We uploaded all relevant documentation, including KTDP expressions, translated OPS procedures, and background materials, to a private GPT environment. For each KTDP entry, we either asked the model to suggest the best-matching OPS procedure directly or first requested an interpretation of the KTDP term followed by a context-based match. This approach allowed us to assess whether ChatGPT's contextual reasoning could complement or outperform traditional embedding-based or lexical matching methods.

3 Evaluation

The absence of a validated ground truth presents a fundamental challenge in assessing the quality of our matching results. Without expert clinical validation, it is unclear how accurate individual matches are or which method performs best. To address this, we first conducted a broad quantitative analysis to evaluate consistency, disagreement, and similarity across methods. These metrics provide indirect but informative insights into model behaviour, helping to characterise matching patterns even in the absence of formal validation. Following this initial assessment, we performed a small-scale manual review to better understand the plausibility of selected matches. We examined examples with both high and low matching scores, identifying cases of clear agreement as well as notable mismatches. This informal inspection offered additional intuition on method performance and highlighted the need for domain expertise to reliably judge alignment quality.

To begin the quantitative evaluation, we examined how often different methods assigned KTDP expressions to the same general procedural category. To do this, we compared the prefixes of the top-1 matched OPS codes across all methods, where the prefix corresponds to the first digit of the OPS code and indicates the high-level category of the procedure (e.g., diagnostic, surgical, therapeutic). This allowed us to assess agreement at a broader level, independent of specific code details.

The results revealed a relatively high degree of consistency: in 64.2% of cases (n = 4000), all methods returned OPS codes with the same prefix, indicating agreement on the general procedural category. In the remaining 35.8% of cases (n = 2231), there was partial agreement - some methods aligned on the prefix, while others diverged. Notably, there were no cases in which all methods assigned entirely different prefixes, suggesting that at least a minimal level of agreement was always preserved at the category level.

However, when comparing full OPS codes, agreement dropped substantially. Only 2.9% of cases (n = 178) exhibited full consensus across all methods. Most cases (90.1%, n = 5613) fell into the "some same" category, where at least two methods agreed, and 7.1% (n = 440) showed complete disagreement, with each method proposing a different code. These results indicate that, while methods often converge on the general category of a procedure, they frequently differ in the specific code they assign within that category.

To further examine how the methods differ in their assignment behaviour, we analysed the distribution of top-1 matched OPS codes across the six main procedural chapters. As illustrated in Figure 2, all methods predominantly mapped KTDP expressions to Chapter 5 (surgical procedures), reflecting the procedural nature of the source data. In contrast, assignments to Chapter 6 (medications) and Chapter 9 (supplementary measures) were relatively infrequent. This general distribution pattern was consistent across methods, indicating a shared tendency to favour procedural codes.

Even so, some notable differences were observed. For example, GatorTron assigned fewer expressions to Chapter 5 compared to the other methods and exhibited a relatively higher proportion of matches to Chapter 8 (non-operative therapeutic measures). Manual review of these cases revealed that many of the expressions lacked a clearly corresponding OPS code, which may have led the model to prefer broader categories. Still, in the absence of expert validation, we cannot determine whether such assignments are more or less accurate.

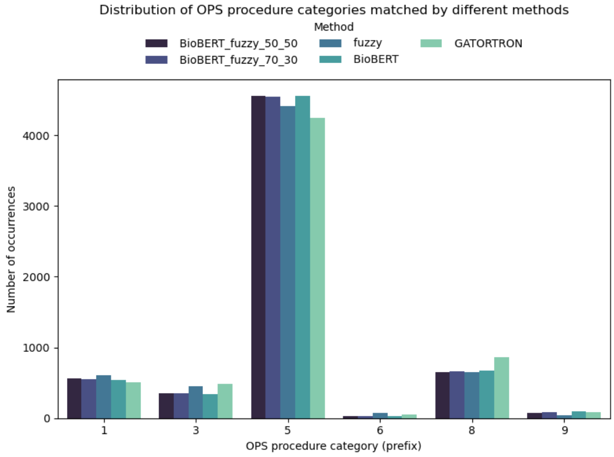


Figure 2: Distribution of top-1 matched OPS codes across the six main procedural chapters for each matching method. Chapter 1 represents diagnostic measures, Chapter 3 imaging diagnostics, Chapter 5 surgical procedures, Chapter 6 medications, Chapter 8 non-operative therapeutic measures, and Chapter 9 supplementary measures.

To investigate whether certain KTDP procedures are inherently easier to match due to wording or alignment with OPS terminology, we analysed the standardised match score values across all methods using a heatmap (Figure 3). The goal was to determine whether consistent scoring patterns could help identify procedures that are generally easier or more difficult to match, regardless of the specific method used.

The heatmap displays Z-standardised scores for each method, with expressions sorted by BioBERT scores. Although we expected some consistency (i.e., easier expressions receiving higher scores across all methods and harder ones receiving lower scores), the results showed considerable variation. In many cases, a procedure scored higher with one method and lower with another, suggesting that matching difficulty is method-dependent and influenced by how each approach interprets textual or structural similarity.

Notably, BioBERT and the hybrid BioBERT-fuzzy method produced very similar score profiles. GatorTron and fuzzy approach showed more divergence, indicating different sensitivities to terminology structure, dataset alignment, or surface-level phrasing.

This suggests that methods differ not only in which codes they select, but also in how confidently they make those matches.

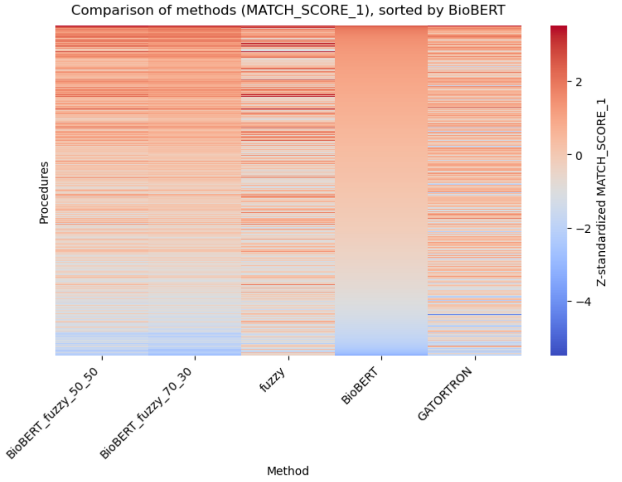


Figure 3: Heatmap of Z-standardised `MATCH_SCORE_1` values across all KTDP expressions, sorted by BioBERT scores. The plot illustrates variation in score strength across methods, highlighting differences in confidence and matching behaviour.

After developing a broader understanding of inter-method differences through quantitative analyses, we conducted a focused manual review of selected examples to qualitatively assess the plausibility of top matches. We examined expressions with both high and low matching scores across methods to explore whether any consistent patterns could be observed.

For expressions with high scores and full agreement across methods, the matches were typically straightforward: the KTDP expression was either identical or highly similar to an OPS entry, often requiring no complex interpretation. These cases tended to represent procedural descriptions that appeared in both datasets with minimal variation.

In contrast, lower-scoring expressions revealed more complex challenges. Two main issues emerged during manual inspection. First, several KTDP procedures had no direct equivalent in the OPS system because they are typically recorded in other coding systems (e.g., vaccinations or disease-specific protocols). Second, many KTDP expressions were written in a general or aggregated form, often combining multiple procedural steps into a single description. OPS, on the other hand, is highly granular, with detailed and precisely defined codes. As a result, some KTDP expressions may correspond to multiple distinct OPS codes, or only partially align with available entries.

These observations suggest that performance limitations are not solely attributable to matching algorithms themselves, but also to structural mismatches and representational differences between the source datasets. This highlights a key challenge in aligning procedural coding systems across countries.

3.1 ChatGPT

Despite leveraging ChatGPT's capacity for contextual reasoning by first interpreting the KTDP expression and then performing the match, the resulting OPS codes were, in most cases, identical to those produced by previously described methods. This suggests that the added interpretation step did not substantially improve matching performance. As previously discussed, this outcome likely reflects the inherent differences in datasets.

4 Conclusion

Our study highlights the considerable challenge of aligning procedural coding systems across countries with different documentation practices. Despite employing a range of computational methods (ranging from fuzzy matching and semantic embeddings to large language models) the observed differences in dataset structure and content significantly limited matching performance. In particular, the lack of detail in some KTDP expressions, the high specificity of OPS codes, and the absence of one-to-one equivalents all contributed to inconsistent or ambiguous results.

Crucially, no ground truth currently exists to objectively evaluate the quality of these matches. Although indirect metrics and manual inspection provide useful information, they cannot replace expert validation. Therefore, the most important next step is to involve medical professionals in generating a gold standard reference set. This would enable formal benchmarking of different methods and support the development of more reliable and generalisable code alignment pipelines in the future.

Ultimately, our findings suggest that the key limitation lies not in the technical capability of the methods themselves, but in the fundamental heterogeneity of the datasets and the differing philosophies of procedural encoding. Addressing this mismatch will be essential for any future efforts to enable international interoperability of procedural coding systems.

Acknowledgments

The authors are grateful for the valuable input and ideas contributed by the medical team of the University Medical Center Ljubljana.

Funding

This work was supported by the Slovenian Research and Innovation Agency (Research Core Funding Number P2-0209).

References

[1] [SW] Max Bachmann, rapidfuzz/RapidFuzz: Release 3.13.0 version v3.13.0, Apr. 2025. DOI: 10.5281/zenodo.15133267, URL: https://doi.org/10.5281/zenodo.15133267.

[2] Bundesinstitut für Arzneimittel und Medizinprodukte (BfArM). 2023. *Operationen- und Prozedurenschlüssel (OPS), Version 2024: Internationale Klassifikation der Prozeduren in der Medizin – Systematisches Verzeichnis*. BfArM. Bonn, Germany.

[3] Pritam Deka, Anna Jurek-Loughrey, et al. 2022. Evidence extraction to validate medical claims in fake news detection. In *International Conference on Health Information Science*. Springer, 3–15.

[4] Marcin Junczys-Dowmunt et al. 2018. Marian: Fast Neural Machine Translation in C++. Tech. rep. arXiv:1804.00344. Demonstration paper, version v3. arXiv, (Apr. 2018). DOI: 10.48550/arXiv.1804.00344.

[5] Bunyamin Keles, Murat Gunay, and Serdar Caglar. 2024. LLMs-in-the-loop Part-1: Expert Small AI Models for Bio-Medical Text Translation. Tech. rep. arXiv:2407.12126. Preprint. arXiv, (July 2024). DOI: 10.48550/arXiv.2407.12126.

[6] Nacionalni inštitut za javno zdravje (NIJZ). 2023. *Klasifikacija terapevtskih in diagnostičnih postopkov in posegov: Pregledni seznam (Verzija 11)*. NIJZ. Ljubljana, Slovenia.

[7] Zabir Al Nazi and Wei Peng. 2024. Large language models in healthcare and medical domain: a review. *Informatics*, 11, 3, 57. DOI: 10.3390/informatics11030057.

[8] OpenAI. 2024. Gpt-4o. Accessed: August 2025. (2024). https://openai.com/index/gpt-4o.

[9] Mohammed Suleiman Mohammed Rudwan and Jean Vincent Fonou-Dombeu. 2023. Hybridizing fuzzy string matching and machine learning for improved ontology alignment. *Future Internet*, 15, 7, 229. DOI: 10.3390/fi15070229.

[10] Xi Yang et al. 2022. A large language model for electronic health records. *npj Digital Medicine*, 5, 1, 194.

AI-Enabled Dynamic Spectrum Sharing in the Telecommunication Sector – Technical Aspects and Legal Challenges

Nina Rechberger
PhD Candidate Applied AI
Alma Mater Europea
Maribor, Slovenia
nina.rechberger@almamater.si

Abstract

Dynamic Spectrum Sharing (DSS), as part of Dynamic Spectrum Management, is already used in the telecommunication sector and is a critical technology for addressing spectrum scarcity in next-generation wireless networks, particularly when implementing 6G. Legacy statical spectrum management (designed for one user exclusively for a certain bandwidth for certain services) is no longer fit for purpose, as it does not allow the efficient use of the spectrum. By leveraging Artificial Intelligence (AI), DSS enables the real-time adaptive allocation of radio frequencies, thereby improving spectrum utilization and network efficiency. Although the integration of AI into DSS introduces complex technical and legal challenges. This paper aims to investigate the challenge of dynamic spectrum policy when using AI-enabled DSS and answer the question of why a flexible and new spectrum policy is desired. Some suggestions for refining the regulatory framework are also presented, which are long overdue in academic research. Recent research primarily focuses on technical issues, rather than specifically on legal ones. The closure findings underscore the need for standardized protocols, adaptive regulatory policies, and other legal frameworks to ensure equitable and efficient spectrum sharing.

Keywords

AI-Enabled Dynamic Spectrum Sharing, AI, spectrum sensing, spectrum right, spectrum regulatory framework

1 Introduction

The integration of Artificial Intelligence (AI) into Dynamic Spectrum Sharing (DSS) introduces technical complexities, such as computational demands and algorithm reliability (e.g., consistency, robustness, and accuracy), alongside legal challenges, including spectrum rights allocation, interference management, and dispute resolution. However, governance frameworks for AI-enabled DSS remain underdeveloped, requiring further exploration.

The rapid growth of wireless devices and data-intensive applications has heightened demand for radio frequency spectrum, a finite resource. Traditional static management often leads to underutilized frequency bands, with inflexible policies exacerbating inefficiencies beyond the spectrum's physical scarcity [1]. AI-enhanced DSS addresses this by enabling flexible, real-time allocation of resources, adapting to dynamic demands and environments while improving spectrum sensing, resource allocation, and interference mitigation.

This study briefly examines the technical and legal dimensions of AI-enabled DSS, identifying challenges and gaps in research. As an initial exploration, it evaluates significant prior work to lay the foundation for future investigations.

2 Technical Aspects of AI-Driven Dynamic Spectrum Sharing

AI-driven DSS leverage all sort of AI techniques to optimize spectrum utilization in dynamic, complex environments. [2, 3, 4].

2.1 Spectrum Sensing and Cognitive Radio

Spectrum sensing is the cornerstone of the DSS, enabling real-time detection of spectrum occupancy. AI-based techniques, such as convolutional neural networks (CNNs) and long short-term memory (LSTM) models, enhance spectrum sensing by analyzing signal patterns and predicting spectrum availability [5, 6]. CNNs are highlighted for their ability to extract features from spectral data, improving detection accuracy in noisy environments without relying on prior knowledge of signals. LSTMs are emphasized for their ability to handle sequential and time-series data..

In addition, deep learning-based spectrum sensing achieves up to 45% improvement in detection accuracy compared with traditional methods, which rely only on basic signal processing techniques to identify spectrum occupancy like energy detection [5]. Cognitive radio networks (CRNs) powered by AI allow users to opportunistically access unused spectrum bands without interfering with other users [3]. The challenges include, among others, the computational complexity of real-time processing and the need for robust datasets to train AI models. Studies highlight that AI models may struggle with unpredictable interference patterns, necessitating hybrid approaches that combine

Information Society 2025, 6–10 October 2025, Ljubljana, Slovenia
© 2025 Copyright held by the owner/author(s).
http://doi.org/DOI_10.70314/is.2025.skui.3657

interpretable models (e.g., decision trees) with high-performing deep learning (DL) models [6].

2.2 Interference Management

Interference management is critical for ensuring reliable connectivity in the DSS. AI-driven techniques, such as multi-agent reinforcement learning (MARL), optimize power allocation and beamforming to minimize interference [6]. MARL is used for mitigating jamming attacks, where malicious entities disrupt spectrum utilization by interfering with communications Another example is reconfigurable intelligent surfaces (RIS) integrated with AI, which can dynamically adjust signal propagation to reduce interference in non-orthogonal CRNs [7]. RIS, also known as an Intelligent Reflecting Surface (IRS), is a passive, planar metasurface composed of a large array of low-cost, tunable unit cells that can dynamically manipulate incident electromagnetic waves. Unlike active devices like base stations or relays, RIS does not generate or amplify signals—it reflects, refracts, or absorbs them in a programmable way to shape the wireless propagation environment.

Research has demonstrated that AI-driven interference management achieves a spectrum utilization efficiency of up to 62.4% in urban environments, nearly double the utilization efficiency compared to traditional management [5]. Although challenges persist, including the scalability of AI models in large networks and the risk of unpredictable behavior in edge cases. Robust fallback mechanisms are necessary to address unpredictable AI behavior in edge cases, while standardized interfaces and protocols are essential for enabling seamless deployment and integration with existing network infrastructure [5].

2.3 Resource Allocation

AI enables dynamic resource allocation by predicting network traffic and allocating spectrum based on real-time demands. Machine learning algorithms, such as support vector machines (SVMs) and deep reinforcement learning (DRL), can forecast spectrum occupancy and optimize bandwidth allocation [8]. For instance, DRL-assisted virtual network embedding (VNE) in satellite networks enhances resource utilization by adapting to multiple coverage constraints [4]. Major obstacles include the need for energy efficiency and the requirement for real-world datasets to enhance prediction accuracy. The absence of standardized testbeds and benchmarks further complicates performance evaluation [2].

3 Regulatory Challenges in AI-Enabled Dynamic Spectrum Sharing

The deployment of AI-driven DSS raises significant regulatory challenges that must be addressed. According to recent research, regulatory issues arise, particularly in interference management, spectrum rights, and dispute resolution. Other legal and regulatory questions have, to the best of the author's knowledge, been completely overlooked or only superficially discussed.

3.1 Interference Management

Interference management in DSS requires regulators to ensure compliance with technical standards to prevent harmful interference. Those standards, when using AI, are missing. An example of such AI-driven systems to avoid interference is spectrum access systems (SAS) that use geolocation databases and sensing to manage the shared spectrum [9]. Simultaneously, the complexity of AI algorithms raises concerns about transparency and accountability when unwanted interference occurs. National regulatory bodies already emphasize the need for standardized protocols to ensure equitable access and interference mitigation [10, 11]. Regulators must strike a balance between innovation and the protection of incumbent users and their guaranteed rights to spectrum.

3.2 Spectrum Rights and Equitable Access

Regulatory authorities adopt the fixed spectrum access (FSA) policy to allocate different parts of the radio spectrum with a certain bandwidth to certain services. With such a static and exclusive spectrum allocation policy, only the authorized users, also known as licensed users, have the right to utilize the assigned spectrum, and the other users are forbidden from accessing the spectrum, regardless of whether the assigned spectrum is busy or not [3]. This could be seen as a direct opposition to the efficient use of the spectrum, where the use of the spectrum aligns with all available technical possibilities. Spectrum rights allocation is a contentious issue in DSS, as AI enables dynamic access by multiple users and challenges traditional licensing models. Spectrum right allocation is traditionally static – one user to a particular broadband. On the other hand, with shared access regimes, such as licensed shared access (LSA), regulators allow spectrum users to open spectrum bands while protecting incumbent users [12]. However, only a few countries have adopted this option, and it comes with numerous regulatory restrictions. For explanation, incumbent users are historically incumbent telecommunications operators, who paid a significant amount of fees for the licence to use the spectrum. Therefore, spectrum licenses are important assets for incumbent users. Nevertheless, AI-driven DSS raises concerns about monopolistic practices because dominant operators may leverage advanced algorithms to secure disproportionate spectrum access [13, 14. 15]. However, legal frameworks must evolve to address equitable access for smaller operators and license-exempt users while simultaneously protecting the guaranteed rights of incumbent users/operators. The absence of clear spectrum rights allocation policies risks exacerbating disputes and stifling innovation in the industry.

3.3 Dispute Resolution

Dispute resolution in DSS tackles conflicts over interference, spectrum access, and user priority. AI systems complicate this due to poor interpretability, obscuring decision processes [6]. AI-driven user prioritization can spark fairness disputes. National spectrum strategies propose interagency resolution processes [6, 10]. Explainable AI models (e.g., XAI) improve transparency, aiding dispute resolution [6]. Blockchain-based databases offer tamper-proof spectrum usage records, simplifying conflict resolution [6].

4 The Need for New AI-Enabled DSS Governance, Suggested New framework

As stated above, traditional regulatory frameworks designed for static spectrum licensing are ill-equipped to handle AI's autonomous and data-intensive nature of AI. The proposed regulatory framework should impose legal mechanisms to address more flexible licensing, privacy and data protection, interference management, security, and international coordination, ensuring compliance and fostering innovation. The objectives of the new framework, in the author's opinion, are: **Enabling Innovation**; **Ensuring Compliance**: that is, aligning with existing laws (e.g. national telecom regulations, Data Act, Artificial Intelligence Act etc.); **Promoting Fairness**, which means ensuring equitable spectrum access and accountability in AI decisions.; **Support Global Harmonization** to align with international standards (e.g., ITU, 3GPP); **Security and Cybersecurity**; Promoting **Regulatory Sandboxes,** to enable safe testing of AI-driven DSS.

4.1 Proposed Legal and Regulatory Framework

4.1.1. Dynamic Licensing Model

Replacing the current policy of static and exclusive spectrum with the Dynamic Licensing Model is a key principle, or, even better, the Dynamic Licensing Model should be prioritized. This could include a tiered access system (primary, secondary, and opportunistic users) managed by AI-driven Spectrum Access Systems (SAS) [3, 12, 9]. This means enacting laws defining tiered access rights, specifying priority levels, and usage conditions. For instance, extending the U.S. Citizens Broadband Radio Service (CBRS) model, where SAS dynamically assigns spectrum, with legal provisions for AI oversight and auditability. Refinements to the European Electronic Communication Code (EECC) [13]. to add AI spectrum management tools are another possible example. First, a definition of DSS should be added and represented. (e,g, in Art. 2). DSS can be defined as a primary shared use of the radio spectrum, enabling flexible, real-time allocation of spectrum bands among multiple users and designated services, when appropriate, adding tiered access rights. In spectrum management (Art. 45 EECC), the goal should also be, by default, to privilege AI-enabled DSS, adding appropriate certification. So, spectrum management could be flexible enough for new technologies and, at the same time, compliant as an exception to the technology and service-neutral principle, traditionally anchored in EECC, because general interest objectives are at stake and can be clearly justified and subject to regular review. From a practical point of view, mandating AI-predictive models for real-time allocation in "AI-harmonized" bands that require shared AI datasets could be discussed in future peer reviews. The neutral authorization regime for spectrum designation, with some exceptions, should move to the explicit inclusion of AI/ML, with possible certification for bias-free algorithms and energy metrics in an additional separate regulation, such as the Gigabyte Infrastructure Act (GIA), intended to simplify access to physical infrastructure in this sector. Art. 46 EECC is meant only to encourage shared access, while the default AI-driven DSS could drive spectrum sharing to another level.

The dynamic licensing model can use blockchain-based smart contracts to automate spectrum allocation, ensuring transparency and enforceability. Regulators should issue guidelines for AI algorithms to prioritize licensed users while optimizing opportunistic/dynamic access and imposing penalties for non-compliance.

4.1.2. Privacy and Data Protection

The goal is to require licensed users to implement privacy-preserving AI techniques (e.g., Federated Learning and differential privacy) to minimize data exposure. Minimal data exposure goes beyond personal data and should be extended to all processed data sets. AI systems in DSS are designed to process only the necessary data for the requested task. Memorized data, such as geolocation and traffic patterns, should be encrypted. Therefore, developing standards for anonymized data processing in DSS, with certification for compliant AI systems, is necessary. For instance, blockchain contracts and differential privacy could enhance efficiency in dense networks and align with the principle of minimizing sensitive data sharing. But on the other hand, all the relevant data for enabling AI-enabled DSS must be shared. Data Act of the EU could address this issue.

Privacy and data protection are strongly connected to the Right to Explanation (transparency). Therefore, it is necessary to mandate transparency in AI-driven spectrum decisions, allowing users to challenge allocations [6, 11]. Although the Artificial Intelligence Act of the EU requires high-risk AI systems (DSS component is legally interpreted as critical infrastructure) to face a strong transparency obligation, in the context of DSS, it needs to be technically detailed.

4.1.3. Interference Management, Liability and Dispute Resolution

Clear liability rules for AI-induced interference, balancing the responsibilities of operators, secondary users, and vendors, must be established. A shared liability model could be a solution. Operators as primary users could be liable for interference unless caused by secondary users or by the vendor/distributor/supplier AI errors, verified through forensic logs. The interference threshold must be introduced and known at the front. Legal limits for acceptable, e.g., signal-to-noise ratio standards, should be defined. The requirement for AI systems to maintain tamper-proof logs of spectrum allocation decisions, accessible when needed to stakeholders, is a good way to ensure the transparent operation of DSS. These logs can then be used as evidence at competent bodies in dispute resolution to resolve interference disputes, with AI decisions [5, 10].

4.1.4. International Standardization

Promoting harmonized standards for AI-driven DSS through international bodies like ITU and 3GPP is just one side of the remaining challenges, like interoperability. Negotiating bilateral and international treaties to align spectrum sharing protocols and data sovereignty rules is another issue. For instance, ITU's World Radiocommunication Conference (WRC) could develop model laws for national adoption, ensuring compatibility with global 5G and 6G standards [10, 14, 15]. Cross-border Coordination (e.g., Art. 4 EECC) could also be expanded, with the RSPG-led cooperation utilizing AI tools for interference resolution.

4.1.5. Security and Cybersecurity

A robust cybersecurity framework for AI-driven DSS systems is aimed at preventing attacks such as data poisoning. Cybersecurity standards for AI-Enabled DSS must still be developed. These standards will include encryption, intrusion detection, and regular security audits for AI systems, as well as reporting security breaches. Certifying AI systems for cybersecurity compliance, with the development of AI-enabled DSS [10, 14, 15].

4.1.6. Regulatory Sandboxes

Creating controlled environments to test AI-driven DSS without full regulatory constraints could be a way to overcome the development compliance. Sandbox legislation should define the scope, duration (e.g., 1-2 years), and liability exemptions for sandbox participants. Launching pilot programs with telecom operators and ensuring legal protections for experimental deployment are essential for the progress of AI-enabled DSS. After the test period, the transition to actual use in the real world would be enhanced because of a good testing foundation in a technological and regulatory sense. A good example is the Model on the UK's Ofcom sandbox, tailored for AI-driven 6G applications [10, 14, 15]. When it comes to regimes for authorization (e.g., Art. 47 EECC), introducing "AI-sandbox" authorizations for DSS testing accelerates innovation through pilots accompanied by authorization. This is also in line with the AI Act, where sandboxes represent well-documented risk mitigation and, as a result, transparency.

5 Conclusion

In this paper, the author examined AI-Enabled DSS from a technical and legal governance perspective. This is a notable achievement because there is a significant gap in research in this field.

This paper aimed to highlight some dimensions of the interaction between technological perspectives and the governance of AI-enabled DSS. After reviewing the adversarial and inherited technical challenges, such as resource allocation, interference management, and spectrum sensing, the legal issues of interference management, spectrum allocation, and equitable access, along with dispute resolution, are briefly discussed.

Moving into the future, a new possible regulatory framework is presented, including a dynamic licensing model, the implementation of privacy-preserving AI techniques in DSS, and a shared liability approach to interference management that could also contribute to dispute resolution. Briefly, the importance of international standardization and interoperability, as well as cybersecurity threats such as data poisoning and the lack of standardization, is mentioned. Lastly, creating regulatory not only technical sandboxes as controled testing environments is proposed.

Acknowledgments

This work was encouraged by the University of Alma Mater Europaea and the Doctoral Program in Applied Artificial Intelligence.

References

[1] Pranita Bhide, Dhanush Shetty, Suresh Mikkili. 2024. Review on 6G communication and its architecture, technologies included, challenges, security challenges and requirements, applications, with respect to AI domain. IET Quantum Communication. **https://doi.org/10.1049/qtc2.12114**

[2] Sabir, Bushra, et. 2024. Systematic Literature Review of AI-enabled Spectrum Management in 6G and Future Networks." arXiv preprint arXiv:2407.10981. https://arxiv.org/abs/2407.10981, https://doi.org/10.48550/arXiv.2407.10981

[3] Ying-Chang Liang 2020 Dynamic Spectrum Management: From Cognitive Radio to Blockchain and Artificial Intelligence, Springer. https://doi.org/10.1007/978-981-15-0776-2

[4] Alhammadi, Abdulrahman et. 2024. Artificial Intelligence in 6G Wireless Networks: Opportunities, Applications, and Challenges. International Journal of Communication Systems.: https://onlinelibrary.wiley.com/doi/10.1002/dac.5443

[5] Saurabh Hitendra Patel. 2024. Dynamic Spectrum Sharing and Management Using Drone-Based Platforms for Next-Generation Wireless Networks. Preprints.org. https://www.preprints.org/manuscript/202412.0854/v2

[6] Abiodun Gbenga-Ilori. 2025. Artificial Intelligence Empowering Dynamic Spectrum Access in Advanced Wireless Communications: A Comprehensive Overview. MDPI. https://www.mdpi.com

[7] Robin Chataut et. 2024. 6G Networks and the AI Revolution—Exploring Technologies, Applications, and Emerging Challenges. PMC. https://pmc.ncbi.nlm.nih.gov/articles/PMC10969307

[8] Mehmet Ali Aygül. 2025. Machine learning-based spectrum occupancy prediction: a comprehensive survey- Frontiers in Communications and Networks. https://www.frontiersin.org/articles/10.3389/frcmn.2024.1345678

[9] Janette, Stewart. 2024. Improved management of shared spectrum: a potential AI/ML use case. Analysys Mason. https://www.analysysmason.com

[10] Anonymus. 2024. Advanced Dynamic Spectrum Sharing Demonstration in the National Spectrum Strategy. National Telecommunications and Information Administration. https://www.ntia.gov/issues/national-spectrum-strategy/advanced-dynamic-spectrum-sharing-demonstration-in-the-national-spectrum-strategy

[11] Anonymous (2025). FCC TAC AI-WG Artificial Intelligence Meeting Slides. https://www.fcc.gov/sites/default/files/08-05-2025-FCC-TAC-Meeting-Slides-Merged.pdf

[12] Anonymous 2025. Spectrum management: Key applications and regulatory considerations driving the future use of spectrum." Digital Regulation Platform. https://digitalregulation.org

[13] Directive (EU) 2018/1972 of the European Parliament and of the Council of 11 December 2018 establishing the European Electronic Communications Code, http://data.europa.eu/eli/dir/2018/1972/oj

[14] Anonymous 2024. Artificial Intelligence in Spectrum Management: Policy and Regulatory Considerations." IEEE Conference Publication. https://ieeexplore.ieee.org

[15] Hussein, Haval 2025. AI-Driven Cognitive Radio Networks for 6G: Opportunities and Challenges. IEEE Transactions on Wireless Communications. https://ieeexplore.ieee.org

SmartCHANGE Risk Prediction Tool: Next-Generation Risk Assessment for Children and Youth

Nina Reščič
Jožef Stefan Institute,
Jožef Stefan International
Postgraduate School,
Ljubljana, Slovenia
nina.rescic@ijs.si

Marko Jordan, Sebastjan
Kramar, Ana Krstevska,
Marcel Založnik
Jožef Stefan Institute,
Jožef Stefan International
Postgraduate School,
Ljubljana, Slovenia

Lotte van der Jagt
Harm op den Akker
Martijn Vastenburg
Research & Development
ConnectedCare
Nijmegen, The Netherlands

Valentina Di Giacomo
Elena Mancuso
Engineering Ingegneria Informatica
SpA
Rome, Italy

Dario Fenoglio
Gabriele Dominici
Università della Svizzera italiana
Lugano, Switzerland

Mitja Luštrek
Jožef Stefan Institute,
Jožef Stefan International
Postgraduate School,
Ljubljana, Slovenia

Abstract

Non-communicable chronic diseases (NCDs), largely driven by lifestyle factors such as poor nutrition, physical inactivity, and obesity, account for over 70% of mortality in Europe. While prevention has traditionally focused on adults, growing evidence highlights the value of early intervention during childhood and adolescence to establish healthy behaviours and reduce long-term risk. This paper presents the updated SmartCHANGE platform, which harmonizes heterogeneous datasets, addresses missing information through synthetic data generation, and forecasts risk factors from childhood to adulthood. Forecasts are then applied to established cardiovascular and diabetes risk models, enabling long-term risk assessment. To ensure privacy, the platform incorporates federated learning for secure model training across distributed datasets. By combining synthetically generated data, predictive modelling, privacy-preserving infrastructure, and end-user applications, the updated SmartCHANGE platform supports early identification of at-risk youth and enables targeted, data-driven interventions to help reduce the future burden of NCDs.

Keywords

non-communicable diseases, risk prediction, synthetic data generation, federated learning, preventive healthcare

1 Introduction

Non-communicable diseases (NCDs), including cardiovascular disease and diabetes, cause over 70% of deaths in Europe [6]. Their onset is shaped by modifiable risk factors such as diet, physical inactivity, obesity, smoking, and alcohol use. While prevention strategies typically target adults, growing evidence highlights childhood and adolescence as critical periods for establishing lifelong health behaviours [5]. Addressing risk early can delay or prevent NCD onset and promote long-term well-being.

In this paper, we described an updated pipeline for predicting NCD risk in young people, building on our previous paper [4]. The new version introduces three advances: (i) broader harmonization of European cohort datasets through refined syntactic and semantic alignment; (ii) improved synthetic data generation that addresses heterogeneity of the datasets; and (iii) evaluation of advanced RNN-based architectures alongside conventional ML models. While the pipeline in the previous paper powered a simple demo, this one is integrated into the SmartCHANGE prototype that enables early identification of at-risk youth and supports the development of tailored preventive strategies. By combining harmonized datasets, predictive modelling, and privacy-preserving methods, it represents a step toward proactive, data-driven public health focused on youth as a critical stage for prevention. In addition, explainable AI was used to generate counterfactuals that support understanding of risk factors, and both web and mobile applications were developed to deliver these insights directly to healthcare professionals, adolescents, and families.

2 Baseline Predictive Approach

The models for forecasting risk factors are trained on seven heterogeneous datasets, none of which contain all the variables needed for risk prediction. The baseline predictive approach includes synthetic data generation and forecasting of individual risk factors from young to older age using various established machine-learning models. These forecast risk factors are then fed into established risk-prediction models to estimate the risk of cardiovascular disease and diabetes.

2.1 Synthetic Data Generation

The synthetic data generation was used to improve data completeness, enhance cross-dataset comparability, and support more robust forecasting and predictive modeling.

2.1.1 Generation of Diet Scores. The risk models required full dietary information, but none of the project datasets contained all the variables needed for diet scores. We therefore used the EUMenu dataset, which includes the complete set of dietary variables. Scores were first calculated for all EUMenu individuals. For project datasets with overlapping dietary or related features, we trained predictive models on EUMenu using only shared variables and generated synthetic diet scores accordingly. Given the task's simplicity and data structure, linear models were applied.

Information Society 2025, Ljubljana, Slovenia
© 2025 Copyright held by the owner/author(s).
https://doi.org/10.70314/is.2025.skui.7226

2.1.2 Generation of Other Data. We generated synthetic values for missing variables by constructing targeted sub-datasets and generating data with supervised learning. Each sub-dataset required core demographics (sex, age, weight, height); rows missing these were discarded to ensure stable baselines. A greedy search selected predictor sets that maximized coverage of missing entries, informativeness beyond demographics, and training sample size. Candidate sets were ranked by Score $= U \times V \times \sqrt{K}$, where U is the number of missing instances covered, V the number of predictors, and K the number of training rows.

For each sub-dataset, Gradient Boosting, Random Forest, and Linear Regression models were trained with k-fold cross-validation and grid search. Validation was assessed with Root Relative Squared Error (RRSE; where RRSE = 0 for perfect predictions, RRSE = 1 for baseline), and the best model generated the missing values. Overlaps were resolved by keeping predictions from the model with a lower RRSE. This process was repeated across variables to expand coverage while minimizing error. Data generation proceeded iteratively: after each pass, synthetic variables were evaluated with RRSE. Variables below a threshold were accepted and treated as ground truth in the next pass, with sub-datasets and models recomputed accordingly. The procedure terminated once no further variables met inclusion or performance plateaued, yielding a consistent. The mean RRSE of synthetic values in the final dataset was 0.795.

2.2 Risk Factor Forecasting

Having generated synthetic data, we constructed a merged dataset with no missing values. This dataset was used to train machine learning (ML) models to forecast health-related risk factors from childhood into adulthood. The predicted values were then applied as inputs to publicly available risk models to estimate the risk of developing NCDs.

We implemented a neural network (NN) with two dense layers (512 and 128 neurons) to capture non-linear patterns. Training used MSE loss, the Adam optimizer, ReLU activations, dropout (0.2), and early stopping. A single NN forecasted all risk factors simultaneously. Training and test data were prepared by generating all younger-to-older age pairs per individual. Inputs included gender, input and target age, and risk factors at the input age; targets were the same risk factors at the target age. This design enabled the model to learn age-progressive changes.

Input–output pairs were split into training, validation, and test sets, with each individual assigned to only one partition to avoid leakage. Stratification by dataset preserved source representation. Features were standardized with scikit-learn's StandardScaler. For comparison, we trained traditional ML models separately per variable: Linear Regression, Ridge Regression, Random Forest, and LightGBM (the latter via the lightgbm library). All models used default parameters and were trained/tested on the same pairs as the NN. Performance was measured with MAE and RRSE. Training used both real and synthetic data, but evaluation was restricted to real data. Input ages ranged from 6–18 years, and target ages from 18–55 years, matching the SmartCHANGE forecasting scope. The mean RRSE of the forecast values was 0.829.

2.3 Risk Models

We focused on two models: the Healthy Heart Score (HHS) for cardiovascular disease and Test2Prevent (T2P) for diabetes risk. Both include lifestyle factors such as physical activity and diet—essential for assessing younger populations and behavioural change—aligning with our goal of early prevention through modifiable risk factors. Using both models balanced clinical reliability with behavioural relevance, enabling a more comprehensive NCD risk assessment.

Our initial approach applied the models at age 55, the maximum forecastable age. This yielded inconsistent outputs: T2P produced 10-year risks (55–65), while HHS produced a 20-year risk (55–75). To resolve this, we instead reported cumulative risks to age 65, the most suitable endpoint given our data. Two strategies were evaluated: non-overlapping intervals and overlapping (hazard-averaging) intervals.

3 Advanced Unified Predictive Approaches

This section introduces advanced forecasting methods designed to work directly on heterogeneous datasets without requiring prior synthetic data generation. Despite their greater sophistication, their accuracy lags behind the more straightforward method that relies on synthetic data generation.

Synthetic data generation and forecasting are trained jointly within a single model, enabling the sharing of representations and feedback. Early layers provide initial estimates for both tasks, while later stages refine them by capturing complex temporal dependencies. Although SmartCHANGE uses only single-year inputs per user, the training dataset includes multi-year records, which reveal broader behavioural patterns.

Before entering the network, variables are normalized using training set statistics. Synthetic values are first generated in a linear block conditioned on age, gender, and BMI. This block consists of two fully connected layers (128 neurons + ReLU, then 21 neurons without activation). Forecasting then adds current age, future age, and gender, and predicts 21 risk factors across ages 6–55. The forecasting block differs by including an additional 128-neuron ReLU layer and more inputs. Forecasting is performed separately for each input year, and if multiple years exist, trajectories are averaged across target ages (e.g., data at 7, 9, and 12 yield three trajectories averaged per year).

This produces a time series of shape (50, 21). Appending masks for observed/synthetic values and gender gives (50, 43). Risk factor trajectories are then refined via a GRU block with bidirectional layers (128 or 21 hidden units) and a final 21-neuron linear layer. Predictions are finally de-normalized back to the original scale. The overall loss is the mean of two MAE terms: imputation and forecasting, with the latter computed only on ground-truth variables in the recorded output year.

The model was evaluated the same way as the one in Section 2.2, with the mean RRSE being 0.907. This is less than the RRSE from Section 2.2, indicating the need for further refinement of the unified approach.

4 Privacy Preservation and Explainability

Privacy Preservation. Within the SmartCHANGE project, health datasets are distributed across multiple countries and institutions. These sensitive data fall under strict regulations (e.g., GDPR), which prohibit cross-border sharing, and new pilot data remain stored locally, reinforcing isolation. Federated Learning (FL) addresses this by enabling collaborative training without moving raw data [3]. Two main challenges arise in deployment: pronounced heterogeneity across sites and residual privacy risks, since shared gradients can still leak information. To mitigate these, we developed distribution-aware, privacy-preserving FL strategies tailored to real-world healthcare [2]. Instead of a single global model, our approach builds compact, differentially private descriptors of each client's data distribution, clustering similar clients to train specialized models. This improves robustness to variability and temporal drift while ensuring fairer predictions, including for underrepresented groups. On the privacy side, model partitioning and communication-efficient aggregation reduce leakage without heavy cryptography by fragmenting gradients and distributing aggregation. Together, these strategies enable scalable, robust, and privacy-preserving FL pipelines for health risk prediction.

Explainability. Beyond predictive accuracy, effective NCD risk assessment must also provide transparent explanations and actionable guidance. For this, we adapt the Counterfactual Concept Bottleneck Model (CF-CBM) [1] to early-life health data. Instead of relying on predefined concepts—often unavailable or inconsistently annotated—our model learns patient feature distributions via a variational autoencoder (VAE), ensuring the latent space captures key generative factors of early-life trajectories. Counterfactuals are then generated following CF-CBM principles: given a patient profile and its predicted risk, the system proposes minimally altered, realistic configurations that would change the outcome. For example, if a child is predicted at high diabetes risk, the model may suggest plausible counterfactual profiles where lifestyle or physiological factors are adjusted to reduce risk. By embedding counterfactual reasoning directly into the pipeline, this approach goes beyond post-hoc interpretability. It both explains which factors drive predictions and identifies how risk can be reduced, offering clinicians and families actionable, personalized strategies for early prevention.

5 Architecture and User Applications

Architecture. The SmartCHANGE platform (Figure 1) is a modular, microservices-based system for AI-driven health interventions in children and adolescents. It integrates the developed predictive pipeline described in the previous sections with secure, scalable, and privacy-preserving technologies, with emphasis on GDPR compliance and explainable AI. Two main client interfaces are provided: the HappyPlant mobile app for families and youth, and a web application for healthcare professionals (HCPs).

Authentication and authorization are handled through the OpenID Connect (OIDC) protocol, with role-based access control and single sign-on. Additional safeguards include encrypted communication, pseudonymization, and immutable audit logging. Together, the SmartCHANGE platform, HappyPlant, and the HCP web interface form an integrated ecosystem for preventive healthcare, uniting advanced technical architecture with user-centered design to deliver effective, scalable, and personalized interventions.

Web Application. The web application for HCPs serves as a clinical dashboard, enabling them to access patient data, assess long-term risk for metabolic diseases (currently diabetes and CVD, although it can be scaled to integrate additional prediction models), and support behaviour change strategies. The interface is structured around a clinically aligned workflow — Consultation, Assessment, and Intervention — mirroring real-world practices.

Mobile Application. While intelligent risk predictions support HCPs in guiding clients, evidence and co-creation results show that simply communicating risks is insufficient for sustainable behaviour change in adolescents and families. The HappyPlant app was designed to address this gap. Rather than focusing on risks, it adopts a playful plant-growth analogy: users care for a virtual plant by completing daily and weekly personalized challenges linked to long-term health goals set by the HCP. The app nudges users towards the most suitable challenges but leaves the final choice to them, supporting autonomy and agency.

To foster long-term engagement, fully grown plants can be placed in the user's Goal Garden, which both showcases past achievements and acts as a reinforcement mechanism. In today's reward-driven context, the Goal Garden also enables saving towards real-life rewards set by parents, further motivating users. The app's design emerged from an extensive co-creation process and iterative validation with users, who responded positively to the analogy, challenge, and reward structure, as well as the aesthetics. Development was kept flexible, with adjustments made to align the app with other SmartCHANGE components.

6 Conclusion

This paper provides a concise description of the SmartCHANGE pipeline, which integrates harmonized datasets, synthetic data generation, federated learning, and explainable AI into a secure platform for early NCD risk prediction and prevention. Through the HappyPlant app and professional interface, these methods are translated into user-centered interventions that support sustainable behaviour change in youth. Detailed descriptions of the individual components will be published separately.

Acknowledgements

This work was carried out as part of the SmartCHANGE project, which received funding from the European Union's Horizon Europe research and innovation program under grant agreement No 101080965. The SLOfit and ACDsi datasets were provided by the University of Ljubljana (courtesy of Gregor Jurak et al.), the LGS dataset was provided by KU Leuven (courtesy of Martine Thomis), AFINA-TE dataset was provided by the University of Porto (courtesy of José Ribeiro), ABCD was provided by VUMC, HELENA dataset was provided by Helena study group (courtesy of Francisco Ortega), and the University of Turku provided the YFS dataset. We are grateful for their support.

References

[1] Gabriele Dominici, Pietro Barbiero, Francesco Giannini, Martin Gjoreski, Giuseppe Marra, and Marc Langheinrich. 2025. Counterfactual concept bottleneck models. In *The Thirteenth International Conference on Learning Representations*. https://openreview.net/forum?id=w7pMjyjsKN.

[2] Dario Fenoglio, Gabriele Dominici, Pietro Barbiero, Alberto Tonda, Martin Gjoreski, and Marc Langheinrich. 2024. Federated behavioural planes: explaining the evolution of client behaviour in federated learning. In *Advances in Neural Information Processing Systems (NeurIPS 2024), Vol. 37*, 112777–112813.

[3] Dario Fenoglio, Daniel Josifovski, Alessandro Gobbetti, Mattias Formo, Hristijan Gjoreski, Martin Gjoreski, and Marc Langheinrich. 2023. Federated learning for privacy-aware cognitive workload estimation. In *Proceedings of the 22nd International Conference on Mobile and Ubiquitous Multimedia (MUM '23)*. ACM, New York, NY, USA, 25–36. DOI: 10.1145/3626705.3627783.

[4] Marko Jordan, Nina Reščič, Sebastjan Kramar, Marcel Založnik, and Mitja Luštrek. 2024. Smartchange risk prediction tool: demonstrating risk assessment for children and youth. In *Slovenska konferenca o umetni inteligenci. Zvezek A: zbornik 27. mednarodne multikonference Informacijska družba - IS 2024 : 10.–11. oktober, Ljubljana, Slovenija = Slovenian Conference on Artificial Intelligence. Vol. A : proceedings of the 27th International Multiconference Information Society - IS 2024.* Ljubljana, Slovenia, 71–74.

[5] K. Pahkala, H. Hietalampi, T. T. Laitinen, J. S. Viikari, T. Rönnemaa, H. Niinikoski, and et al. 2013. Ideal cardiovascular health in adolescence: effect of lifestyle intervention and association with vascular intima-media thickness and elasticity (the special turku coronary risk factor intervention project for children [strip] study). *Circulation*, 127, 18, (May 2013), 2088–2096.

[6] World Health Organization. 2018. Global Health Estimate 2016: Deaths by Cause, Age, Sex, by Country and by Region, 2000-2016. World Health Organization.

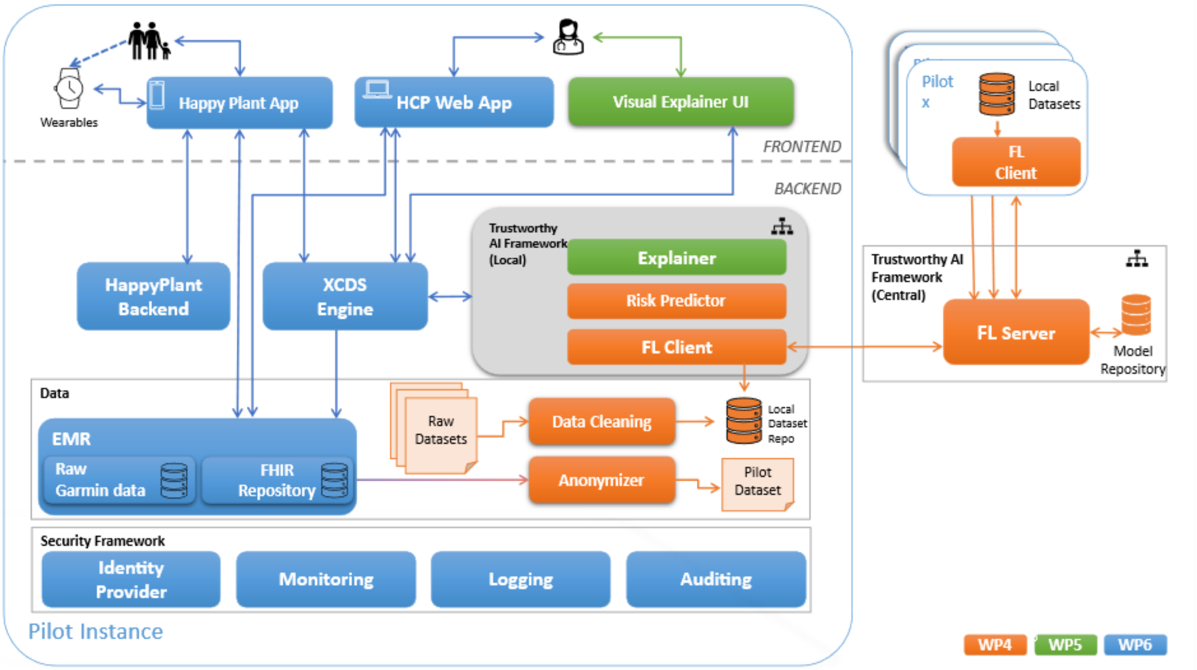


Figure 1: Logical Architecture of the SmartCHANGE Platform, including the mobile app (HappyPlant) and the web-app for healthcare professionals, connected to a central FHIR-compliant repository and featuring a Trustworthy AI Framework with federated learning, explainability, and secure communication via the XCDS Engine.

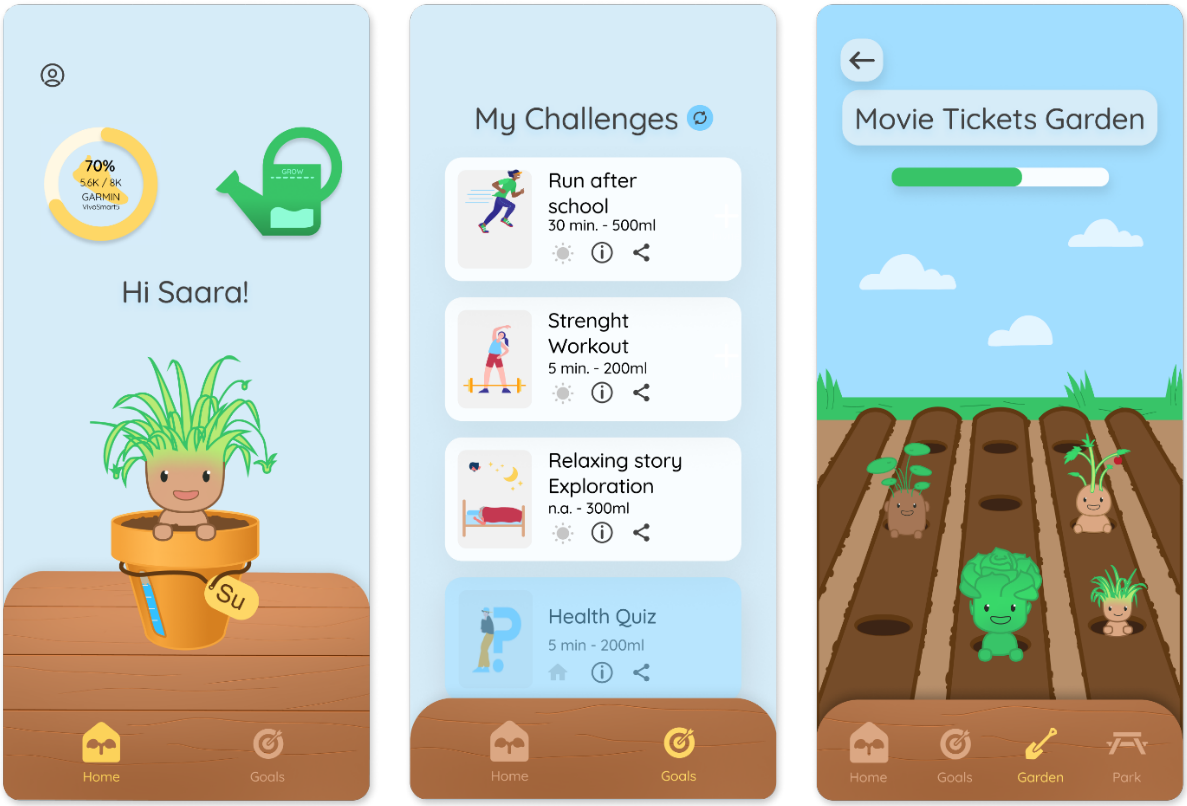


Figure 2: HappyPlant app screens: the home, challenge and goal garden screens.

GNN Fusion of Voronoi Spatial Graphs and City–Year Temporal Graphs for Climate Analysis

Alex Romanova
Independent Researcher
McLean, VA, USA
sparkling.dataocean@gmail.com

Abstract

We present a two-stream graph framework for climate similarity that fuses geography with long-term dynamics. A globe-spanning Voronoi network links cities whose cells share a boundary, while per-city temporal graphs encode decades of daily temperatures in 1000 cities over 40 years. We learn (i) temporal embeddings via a GNN graph-classification model on city–year graphs and (ii) spatial embeddings via a GNN link-prediction model on the Voronoi backbone, using either raw climatology vectors or the learned temporal embeddings as inputs. Treating cosine similarity as edge weights (using 1-cosine) enables graph-mining views: closeness maps highlight dense climate belts, and betweenness maps surface long-range "bridges" connecting distant regions. The fused approach uncovers patterns that simple averages miss, including nearby cities with low similarity (microclimates, urban form, or data aliasing) and far-apart cities with high similarity (shared seasonal regimes/latitude bands). We also incorporate the Delaunay triangulation - the dual of Voronoi - to provide a geometrically well-posed neighbor network that stabilizes these patterns. The method is scalable and reproducible, and the same template - spatial adjacency + temporal history + GNN fusion - extends beyond temperature to additional variables and to urban and infrastructure applications.

Keywords

graph neural networks, spatiotemporal modeling, climate analysis, Voronoi tessellation, Delaunay triangulation

1 Introduction

Understanding global climate patterns is critical to the climate–change challenge. In this study, we explore a graph-based framework that integrates geographic layout with long-term temporal behavior.

As a data source, we use climate records for 1,000 of the world's most populated cities with 40 years of daily temperatures. This dataset (Kaggle [7]) provides geolocations and multi-decade time series, allowing us to combine spatial and temporal perspectives.

Our spatial backbone is a Voronoi graph: from city coordinates, each city receives a Voronoi cell (the region closer to that city than to any other), and two cities are connected when their cells share a border—an interpretable, globally consistent notion of proximity. Alongside Voronoi, we also construct the Delaunay triangulation over the same points. Delaunay provides a complementary, dual view of neighborhood structure and enables triangle-based analyses; we use it as a robustness check to ensure results are not tied to a single choice of spatial adjacency.

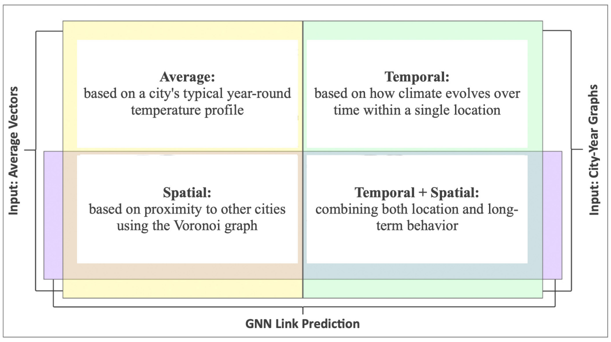

Average:
based on a city's typical year-round temperature profile
Temporal:
based on how climate evolves over time within a single location
Spatial:
based on proximity to other cities using the Voronoi graph
Temporal + Spatial:
combining both location and long-term behavior
Input: Average Vectors
Input: City-Year Graphs
GNN Link Prediction

Figure 1: Node feature types for climate similarity.

For temporal behavior, each city is represented by a graph whose nodes are city–year pairs with daily-temperature profiles as features. Years are linked when their profiles exceed a cosine-similarity threshold. We add a virtual node so that each city graph forms a single connected component.

To analyze climate across space and time, we use basic vectors and pre-final vectors from Graph Neural Network (GNN) models. Figure 1 illustrates four representations used throughout the paper:

- *Average* — climatology vectors (365-day averages) per city;
- *Temporal* — embedded city graphs: pre-final vectors from a GNN graph classification model on per-city year graphs;
- *Spatial* — embedded Voronoi nodes: pre-final vectors from a GNN link-prediction model on the Voronoi graph with average vectors as inputs;
- *Spatial+Temporal* — re-embedded nodes: pre-final vectors from a GNN link-prediction model on the Voronoi graph using temporal embeddings as inputs.

We previously introduced the use of pre-final vectors from a GNN graph classification model on city temporal graphs [17] and applied linear-algebra analyses to those outputs.

In this study we contribute:

- Construction of a globe-spanning Voronoi spatial graph *and* its Delaunay triangulation as complementary spatial backbones;
- Comparisons across input climatology vectors, output city-graph embeddings, and spatial node embeddings from link prediction;
- Graph-mining analyses on induced graphs from each vector type, highlighting agreements and differences across spatial and temporal representations.

Information Society 2025, Ljubljana, Slovenia

https://doi.org/10.70314/is.2025.skui.1600

2 Related Work

In 2012, two milestones reshaped AI: AlexNet's convolutional neural network set a new benchmark in large-scale image classification, far surpassing prior methods [9, 12], and Google's Knowledge Graph operationalized entity–relationship understanding at web-scale, transforming data integration, search, and management [15].

These lines of work initially evolved in parallel—CNNs excelled on grid-structured data, while graph methods targeted relational structure. The emergence of graph neural networks (GNNs) in the late 2010s bridged this gap by combining deep learning with graph computation to model complex dependencies [2]. Despite the rise of large language models (LLMs) since 2022, GNNs remain essential for tasks grounded in explicitly graph-structured data.

GNNs are now standard for classification and link prediction on graph-structured data [14, 1]. At web scale, industrial recommender systems adopt scalable inductive variants such as PinSage [20], while temporal/dynamic settings leverage trajectory-predictive embeddings like JODIE [10]. Community benchmarks have further standardized evaluation for large graph learning (e.g., OGB) [5]. In geophysics, recent studies demonstrate the effectiveness of GNNs for medium-range global weather forecasting [11], global atmospheric prediction [8], and spatiotemporal hydrology and geoscience tasks such as groundwater dynamics [19] and frost-event forecasting with attention mechanisms [13], supporting the view that graph-based inductive biases are well suited to environmental systems with strong spatial and temporal structure.

Voronoi tessellations provide natural adjacency via shared cell boundaries and have a long history in climate and global modeling [6]. Recent applications use Voronoi-induced graphs for urban risk modeling and natural hazards: Gan et al. propose a Voronoi-based spatiotemporal GCN for traffic crash prediction [3], while Razavi-Termeh et al. leverage Voronoi entropy in flood susceptibility mapping [16]. Our work synthesizes these ideas by constructing a global Voronoi-based spatial graph of cities enriched with long-term temperature signals and combining it with per-city temporal graphs encoded by GNNs.

3 Methods

3.1 Graph Construction

We construct a global spatial graph by computing a planar Voronoi diagram on Web Mercator (EPSG:3857) city coordinates; two cities are adjacent if their cells touch. The Voronoi/Delaunay is used only to define adjacency (not distances/areas), yielding a simple, interpretable map of city neighborhoods worldwide.

We evaluate four alternative node-feature sets:

(1) **365-day climatology vectors** — for each city, a 365-value day-of-year climatology averaged across all available years.
(2) **Temporal vectors** — pre-final embeddings from GNN graph-classification model on each city's year-by-year graph (years linked when their daily profiles exceed a cosine-similarity threshold).
(3) **Link-prediction vectors (from averages)** — pre-final embeddings from a GNN link-prediction model on the Voronoi graph using the 365-day climatology vectors as inputs.

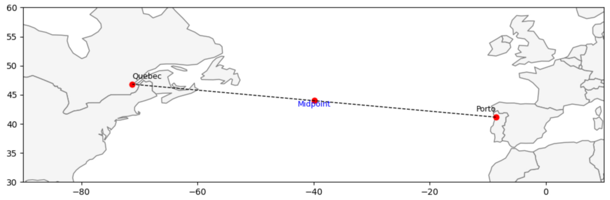


Figure 2: Voronoi edge between distant cities: Québec and Porto are neighbors because their cells meet across the Atlantic.

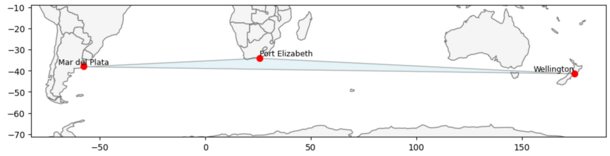


Figure 3: Largest Voronoi triangle: Wellington–Port Elizabeth–Mar del Plata illustrates long edges formed in sparsely populated regions.

(4) **Link-prediction vectors (from temporal vectors)** — the same GNN link-prediction setup, but with temporal GNN embeddings as inputs.

This design allows direct comparison of spatial, temporal, and hybrid representations within a single framework; see Figure 1.

3.2 GNN Graph Classification Model

We apply a GNN graph classification model (PyTorch Geometric) to per-city temporal graphs. Each graph has one node per year, with that year's daily-temperature profile as the node features. We add a virtual node to each graph and connect it to ensure every city graph is a single connected component. For supervision, we split cities into two equal groups by absolute latitude (closer vs. farther from the equator) and train the model to classify the graphs. We then use the pre-final vector as the city's temporal embedding for downstream analysis.

3.3 GNN Link Prediction Model

We apply a GNN link prediction model (Deep Graph Library), using the GraphSAGE aggregator [4], to the Voronoi spatial graph of cities. Unlike the GNN graph classification model, which produces one embedding per city graph, link prediction runs on the global spatial graph and refines each city's node representation using both adjacency and input features. We evaluate two node-feature variants: (i) 365-day climatology vectors (averaged across years) and (ii) temporal embeddings from the classification model. After training, we extract pre-final node embeddings as enhanced city feature vectors for downstream analysis.

Notes and code are provided on our technical blog [18].

4 Experiments

4.1 Voronoi Graph Construction

We build the spatial graph from city coordinates with a Voronoi tessellation: each city gets a cell, and two cities are linked when their cells touch. This gives a clear, globe-spanning picture of who is naturally close, without picking an arbitrary distance cutoff. Alongside this, we also use the Delaunay triangulation

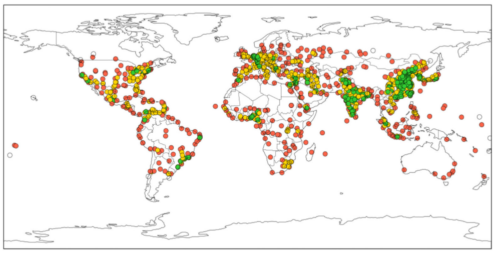


Figure 4: Voronoi area (normalized): green=low, yellow=mid, red=high.

on the same points—the dual view that connects cities exactly when their Voronoi cells meet and highlights triangle-based local structure.

Sometimes this setup links places that are far apart because there are few large cities between them. For example, Québec (Canada) and Porto (Portugal) become neighbors across the Atlantic when their cells meet (Figure 2). Larger patterns show up in the Delaunay view as well: the largest triangle—Wellington (New Zealand), Port Elizabeth (South Africa), and Mar del Plata (Argentina)—illustrates how isolated regions can still form direct connections (Figure 3).

To show spatial density, we color each city by Voronoi cell size (Figure 4). Small cells (green) mark tight clusters—for example, parts of eastern China and northern India—while large cells (red) indicate sparse areas such as interior Australia or northern Canada. Dense hubs shorten edges and raise local connectivity; sparse zones create longer links that act as bridges.

4.2 GNN Models

Across both GNNs (temporal graph classification and spatial link prediction), we use only pre-final embeddings for downstream analysis; we do not report task metrics (edge AUC/AP or classification accuracy) because our goal is weighted-path/centrality analysis on a geometric prior.

4.3 How Similar Are Distant or Nearby Cities?

This section examines climate similarity for both distant and neighboring city pairs using the four representations (*Average*, *Temporal*, *Spatial*, *Spatial+Temporal*). Tables 1 and 2 highlight highlight representative examples: one for geographically distant pairs and one for nearby pairs.

Many distant pairs show very high similarity, especially when temporal history and spatial context are both considered. For example, Wellington (New Zealand) and Mar del Plata (Argentina), though thousands of kilometers apart, score highly across all four metrics—suggesting that similar seasonal regimes and latitude can outweigh raw distance.

Nearby pairs typically agree across metrics as well. In the second table, examples such as Barranquilla–Soledad and Barcelona–Puerto La Cruz show consistently high similarity, reflecting shared local climate.

There are exceptions. New York and Brooklyn, despite being only a few kilometers apart, score low on the *Spatial* and *Spatial+Temporal* measures. This may reflect microclimates, urban effects, or dataset/aliasing issues (e.g., borough vs. city records). Such cases show that short geographic distances can mask meaningful environmental differences, underscoring the value of combining temporal and spatial modeling.

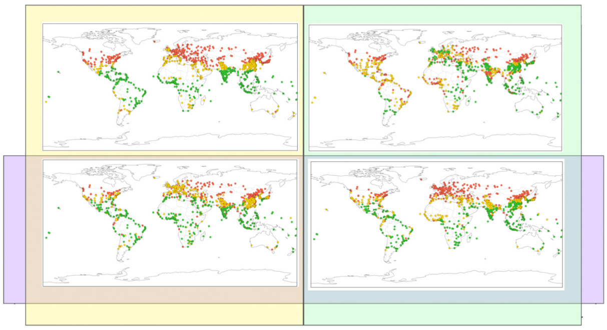


Figure 5: Closeness centrality across four vector types; red = high, yellow = mid, green = low.

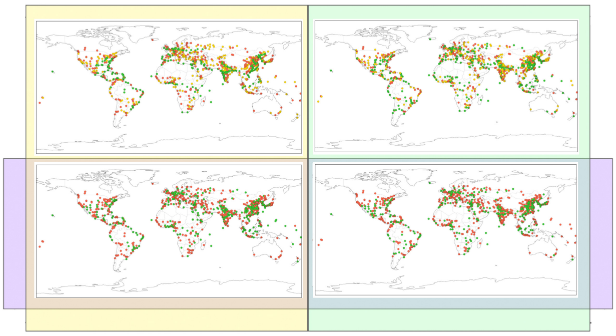


Figure 6: Betweenness centrality across four vector types; red = high, yellow = mid, green = low.

4.4 Centrality and Betweenness Patterns Across Vector Types

Throughout, *climate similarity* means *cosine similarity* between the indicated vectors; for path-based metrics we use edge weights $w = 1 - \text{cosine}$. Each set of maps uses the same spatial backbone: edges come from the Voronoi graph, where two cities are adjacent if their cells share a border. What changes across panels is the edge weight, derived from cosine similarity computed from one of four representations (*Average*, *Temporal*, *Spatial*, *Spatial+Temporal*), with vectors normalized prior to cosine. The topology stays fixed; the weights—and therefore any shortest-path–based measures—change with the chosen vectors. Smaller weights mean higher climate similarity.

In the closeness centrality maps (Figure 5), cities with high closeness are, on average, at short weighted distance from many others—i.e., they are similar to many cities. Dense climate regions such as Europe and East Asia typically stand out. Differences between panels reveal how each representation defines "similar," shifting which cities appear most central.

In the betweenness maps (Figure 6), different weightings emphasize different connectors: high-betweenness cities lie on many shortest routes. The *Spatial+Temporal* view surfaces more long-range intermediaries than *Average* (notably in Africa, South America, and the Pacific). We also observe slight polarization in *Spatial* and *Spatial+Temporal*; the reason for this requires further research.

Our centrality and betweenness maps are only a starting point, with extended graph experiments expected to uncover additional structures and recurring pathways.

Table 1: Climate similarity between distant city pairs

City 1	City 2	Distance (km)	Average	Temporal	Spatial	Spatial+Temporal
Wellington, NZ	Mar del Plata, AR	25870.97	0.9922	1.0000	1.0000	1.0000
Port Elizabeth, ZA	Wellington, NZ	16639.04	0.9982	0.9963	0.9999	1.0000
Melbourne, AU	Port Elizabeth, ZA	13299.30	0.9916	0.9958	0.9872	0.9993
Reykjavik, IS	Krasnoyarsk, RU	12911.14	0.7375	0.7482	0.9861	0.9338
Nuku'alofa, TO	Concepcion, CL	11549.31	0.9838	0.9882	0.9995	0.9997

Table 2: Climate similarity between nearby city pairs

City 1	City 2	Distance (km)	Average	Temporal	Spatial	Spatial+Temporal
Jerusalem, IL	Al Quds, PS	2.27	1.0000	1.0000	0.9998	1.0000
Barranquilla, CO	Soledad, CO	5.63	1.0000	0.9585	0.9999	0.9999
Barcelona, VE	Puerto La Cruz, VE	6.32	1.0000	1.0000	1.0000	1.0000
Khartoum, SD	Omdurman, SD	6.88	1.0000	0.8749	0.9590	0.9988
New York, US	Brooklyn, US	7.05	1.0000	0.5220	0.0857	0.0878

5 Conclusion

In conclusion, the novelty of this work is the explicit fusion of a Voronoi spatial graph with temporal GNN embeddings to reveal climate "neighborhoods" that traditional, single-view methods tend to miss. By running a GNN graph-classification model on per-city year graphs and a GNN link-prediction model on the global Voronoi backbone, we combine geography with long-term dynamics. We compare simple average-by-day climatology vectors against pre-final vectors from both GNN models and then use these vectors for downstream analysis.

This fusion surfaces informative outliers: nearby cities with low cosine similarity—consistent with microclimates, urban form, or data aliasing—and distant city pairs with high similarity, suggesting long-distance climate links. Using these vectors as edge weights enables graph-mining views: closeness maps highlight dense climate belts, while betweenness maps elevate long-range "bridges." Adding the Delaunay triangulation—the dual of the Voronoi diagram—provides a geometrically well-posed neighbor network that stabilizes these patterns.

While this study centers on climate and temperature, the dual Voronoi–Delaunay framework with GNN fusion is broadly applicable. The same geometric scaffold can analyze urban connectivity and infrastructure networks, surface social or economic linkages in dense regions, and support practical tasks like traffic management and siting of schools, parks, or grocery stores. It offers a stable way to reason about spatial relationships beyond climate. The approach is also a starting point for continued work: enrich node features, adopt spherical/geodesic tessellations, learn the graph via contrastive or metric objectives, and explore dynamic temporal GNNs with attribution, counterfactuals, and uncertainty.

References

[1] Jakub Adamczyk. 2022. Application of graph neural networks and graph descriptors for graph classification. *arXiv preprint arXiv:2211.03666*. doi:10.48550/arXiv.2211.03666.

[2] Michael M. Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. 2021. Geometric deep learning: grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*. doi:10.48550/arXiv.2104.13478.

[3] Junjie Gan, Qing Yang, Dong Zhang, Li Li, Xinyu Qu, and Bin Ran. 2024. A novel voronoi-based spatio-temporal graph convolutional network for traffic crash prediction considering geographical spatial distributions. *IEEE Transactions on Intelligent Transportation Systems*. doi:10.1109/TITS.2024.3452275.

[4] William L. Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*. https://arxiv.org/abs/1706.02216.

[5] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. 2020. Open graph benchmark: datasets for machine learning on graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*. https://arxiv.org/abs/2005.00687.

[6] Lili Ju, Todd Ringler, and Max Gunzburger. 2011. Voronoi tessellations and their application to climate and global modeling. In *Numerical Techniques for Global Atmospheric Models*. Lecture Notes in Computational Science and Engineering. doi:10.1007/978-3-642-11640-7_10.

[7] Kaggle Dataset. 2020. Temperature history of 1000 cities 1980 to 2020. https://www.kaggle.com/datasets/sudalairajkumar/daily-temperature-of-major-cities. (2020).

[8] Ryan Keisler. 2022. Forecasting global weather with graph neural networks. *arXiv preprint arXiv:2202.07575*. doi:10.48550/arXiv.2202.07575.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*. doi:10.1145/3065386.

[10] Srijan Kumar, Xikun Zhang, and Jure Leskovec. 2019. Predicting dynamic embedding trajectory in temporal interaction networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*. doi:10.1145/3292500.3330895.

[11] Rosalia Lam, Alvaro Sanchez-Gonzalez, Matthew Willson, et al. 2023. Learning skillful medium-range global weather forecasting. *Science*. doi:10.1126/science.adi2336.

[12] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature*. doi:10.1038/nature14539.

[13] Hugo Lira, Luis Martí, and Nayat Sanchez-Pi. 2022. A graph neural network with spatio-temporal attention for multi-source time series data: an application to frost forecast. doi:10.3390/s22041486.

[14] Xia Liu, Jie Chen, and Qingsong Wen. 2023. A survey on graph classification and link prediction based on gnn. *arXiv preprint arXiv:2307.00865*. doi:10.48550/arXiv.2307.00865.

[15] Natasha F. Noy, Yuval Gao, Anshu Jain, Anant Narayanan, Alan Patterson, and Jamie Taylor. 2019. Industry-scale knowledge graphs: lessons and challenges. *acmqueue*. doi:10.1145/3329781.3332266.

[16] S. Vahideh Razavi-Termeh, Amir Sadeghi, Faisal Ali, Rana Abdul Naqvi, et al. 2024. Cutting-edge strategies for absence data identification in natural hazards: leveraging voronoi-entropy in flood susceptibility mapping with advanced ai techniques. *Journal of Hydrology*. doi:10.1016/j.jhydrol.2024.132337.

[17] Alex Romanova. 2024. Utilizing pre-final vectors from GNN graph classification for enhanced climate analysis. In *Proceedings of the 21st Workshop on Mining and Learning with Graphs (MLG 2024)*. Co-located with ECML PKDD 2024.

[18] sparklingdataocean.com. [n. d.] Temporal–spatial gnn fusion for climate analytics. http://sparklingdataocean.com/2025/06/25/voronoiGNN/.

[19] Marco L. Taccari, Hua Wang, James Nuttall, Xue Chen, and Peter K. Jimack. 2024. Spatial-temporal graph neural networks for groundwater data. doi:10.1038/s41598-024-75385-2.

[20] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L. Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*. doi:10.1145/3219819.3219890.

Towards Anomaly Detection in Forest Biodiversity Monitoring: A Pilot Study with Variational Autoencoders

David Susič
david.susic@ijs.si
Department of Intelligent Systems,
Jožef Stefan Institute
Ljubljana, Slovenia

Maria Luisa Buchaillot
Fauna Smart Technologies ApS
Copenhagen, Denmark

Miguel Crozzoli
Intelligent Instruments Lab,
University of Iceland
Reykjavik, Iceland

Calum Builder
Fauna Smart Technologies ApS
Copenhagen, Denmark

Sevasti Maistrou
Fauna Smart Technologies ApS
Copenhagen, Denmark

Anton Gradišek
Department of Intelligent Systems,
Jožef Stefan Institute
Ljubljana, Slovenia

Dragana Vukašinović
Fauna Smart Technologies ApS
Copenhagen, Denmark

Abstract

Biodiversity monitoring in forests requires scalable, automated tools for detecting ecological anomalies across time and space. This paper reports on a three-month pilot deployment (April 1 to June 30, 2025) in Dyrehaven, an 11 km^2 forest park near Copenhagen, Denmark, where acoustic data from 10 distributed AudioMoth sensors and vegetation indices from Sentinel-2 imagery were collected. We trained separate variational autoencoder (VAE) models on each modality to test the technical feasibility of learning ecological baselines. Since no ecological anomalies occurred during the observation period, evaluation focused on reconstruction errors, which indicate how well VAEs can capture typical site-specific ecological patterns (i.e., baseline modeling). Both acoustic and satellite pipelines achieved low reconstruction errors, demonstrating that VAEs can reliably model normal ecological dynamics. This establishes the foundation for future studies on anomaly detection, which will require larger datasets containing true ecological anomalies identified and labeled by experts. Ongoing work focuses on extending data collection to additional forest sites, while future anomaly detection will require expert-labeled anomalies to calibrate baselines and validate model performance for robust, multimodal biodiversity monitoring.

Keywords

biodiversity, anomaly detection, variational autoencoder, machine learning, passive acoustic monitoring, satellite imagery

1 Introduction

Forests are complex, dynamic ecosystems increasingly affected by environmental stressors such as pests, diseases, invasive species, and climate-related disturbances [1]. Effective biodiversity monitoring is essential to detect these stressors early and support adaptive, science-based forest management [2, 3]. However, existing monitoring tools are often limited in scope, fragmented across disciplines, and costly to implement at scale [4].

This paper presents the technical foundation of the biodiversity assessment tool (BAT), a modular, scalable system that integrates ecoacoustics, satellite remote sensing, and machine learning (ML) to enable automated biodiversity monitoring in forested landscapes. BAT is designed to detect anomalies in ecological baselines, providing early warning signals of ecosystem degradation [5]. It combines two complementary remote sensing modalities: passive acoustic monitoring (PAM), which captures localized, high-frequency biological activity such as insect or bird calls [6, 7], and satellite Earth observation (EO), which offers broader, lower-frequency indicators of landscape-level change, including vegetation health and canopy dynamics [8].

The presence of pests or other stressors often leads to a reduction in biodiversity, which can first be detected acoustically as diminished biotic sound activity, and later (typically with a lag of several days) becomes visible in EO data as decreased vegetation greenness. BAT is designed to leverage this temporal and spatial complementarity by developing independent anomaly detection pipelines for each modality, which in future iterations may support joint multimodal detection of ecological disturbances.

This study reports on a pilot deployment in Dyrehaven, a human-managed park-forest in Denmark, where time-series data from distributed acoustic sensors and Sentinel-2 satellite imagery were collected between April and June 2025. Separate variational autoencoders (VAEs) were trained on each modality to test whether robust baseline models can be learned. Ecological anomalies are inherently rare and cannot be guaranteed within a limited three-month window, and none occurred during this period. As a result, evaluation focused on baseline reconstruction performance rather than anomaly detection accuracy. Demonstrating that VAEs can successfully capture "normal" ecological patterns is a necessary prerequisite for future anomaly detection. Ecological baselines are inherently site-specific, differing across forest types, microhabitats, and even within single forests (e.g., wetter zones near ponds vs drier uplands). Accordingly, this work should be understood as a technical feasibility study, with the longer-term goal of enabling multimodal detection of ecological disturbances such as pest outbreaks, supported by expert-labeled events and extended deployment across diverse forests.

Information Society 2025, Ljubljana, Slovenia

https://doi.org/10.70314/is.2025.skui.6757

2 Data

Our study area was Dyrehaven, a human-managed forest park north of Copenhagen, Denmark (55.8024°N, 12.5685°E), covering 11 km^2 (see Figure 1). The site includes 10 structured microhabitats across woodland, meadow, and modified forest areas. Its ecological diversity and relative stability make it suitable for testing acoustic and satellite-based monitoring methods. Data were collected between April 1 and June 30, 2025.

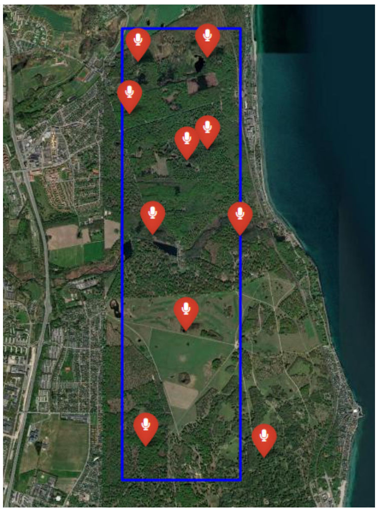


Figure 1: Study area in Dyrehaven, Denmark with AudioMoth recording locations (red pins) and Sentinel-2 satellite bounding box (blue).

2.1 Audio

Passive acoustic data were collected using 10 AudioMoth recording devices deployed across Dyrehaven's microhabitats. Devices were positioned to maximize spatial heterogeneity, minimize acoustic overlap, and ensure temporal consistency. Each unit recorded 45-second mono-channel clips every five minutes at a 48 kHz sampling rate. All devices were weatherproofed and mounted on trees for continuous outdoor operation. A recording gap occurred between April 20 and April 29 due to memory card failure. A total of 203078 recordings were generated during the study period. After removing corrupted or incomplete files (309 clips, 0.15%), 202769 valid recordings remained.

2.2 Visual

Satellite imagery was sourced from the Sentinel-2 mission [9], covering a 1.48 km × 5.86 km bounding box encompassing 9 of the 10 AudioMoth locations. Out of 53 total available snapshots during our study period, 18 cloud-free scenes (≤50% cloud cover) were selected for analysis to ensure index reliability.

Normalized difference vegetation index (NDVI) and Normalized difference moisture index (NDMI) were computed for each selected image as

$$\text{NDVI} = \frac{\text{NIR} - \text{red}}{\text{NIR} + \text{red}}$$

and

$$\text{NDMI} = \frac{\text{NIR} - \text{SWIR}}{\text{NIR} + \text{SWIR}},$$

where, NIR, SWIR, and red are near-infrared, shortwave-infrared, and visible red bands, respectively.

NDVI was calculated at 10 m resolution, and NDMI at 20 m. Each index map was divided into fixed-size patches. NDVI maps produced 396 patches (11 × 36 grid), while NDMI produced 108 patches (6×18 grid), reflecting their respective spatial resolutions.

3 Methodology

3.1 Extraction of Acoustic Indices

10 standard ecoacoustic indices [10] (list in Table 1) were extracted from each 45-second recording, capturing patterns from both time-domain and time-frequency analyses. These indices reflect aspects such as spectral entropy, acoustic complexity, temporal dynamics, and frequency distribution, offering proxies for ecological features like species richness, biophonic activity, and anthropogenic disturbance. All indices were independently normalized to the [0, 1] range using their dataset-wide minimum and maximum values.

Table 1: Acoustic indices used in this study and their ecological interpretation.

Index	Use
ACI	Detects dynamic biotic sounds (e.g., bird choruses).
AEI	Identifies dominance vs. diversity in acoustic communities.
EAS	Differentiates uniform noise vs. structured signals.
ECU	Indicates unpredictability and complexity of soundscapes.
ECV	Captures temporal structure (e.g., insect or bird rhythms).
EPS	Distinguishes tonal vs. noisy sound environments.
ADI	Proxy for acoustic diversity or species richness.
NDSI	Separates natural from human-made noise.
Ht	Detects continuous vs. discrete acoustic events.
ARI	Estimates overall acoustic richness.

3.2 Preprocessing of Satellite Imagery

To ensure patch-level data quality, we applied the scene classification layer (SCL) after resampling. Patches containing cloudy or unreliable pixels (SCL classes 3, 8, 9, or 10) were excluded. This preprocessing pipeline produced curated spatiotemporal datasets of 4436 NDVI patches and 1226 NDMI patches, which served as input for training and evaluating the VAE models.

3.3 Variational Autoencoder and Evaluation Metrics

A variational autoencoder (VAE) learns to compress input data into a latent representation and reconstruct it via encoder and decoder as per Figure 2.

The encoder maps each input to a latent mean, μ_1 and log-variance, $log(\sigma_1^2)$, from which a latent vector z is sampled via the reparameterization trick: $z = \mu_1 + \sigma_1 \cdot \epsilon$, where $\epsilon \sim \mathcal{N}(0, 1)$ and $\sigma_1 = \exp(0.5 \cdot \log(\sigma_1^2))$.

The decoder reconstructs the input from z, producing a mean μ_2 and log-variance $\log(\sigma_2^2)$ of the output distribution. Training minimizes the total loss:

$$\mathcal{L}_{\text{VAE}} = \mathcal{L}_{\text{recon}} + w_{\text{KL}} \cdot \mathcal{L}_{\text{KL}}$$

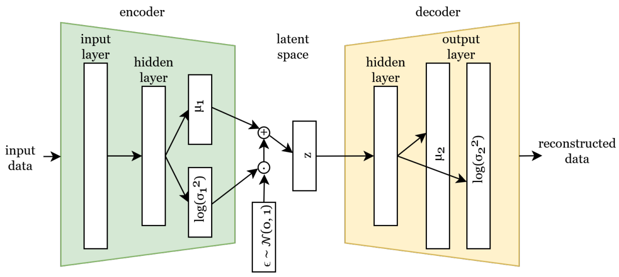


Figure 2: Architecture of VAE for anomaly detection using reconstruction probability.

where $\mathcal{L}_{\text{recon}}$ is the negative log-likelihood of the input under the decoder's Gaussian output:

$$\mathcal{L}_{\text{recon}} = -\sum_{i=1}^{D} \log \mathcal{N}(x_i \mid \mu_{2,i}, \sigma_{2,i}^2)$$

and $\mathcal{L}_{\text{KL}}$ is the Kullback–Leibler divergence between the approximate posterior $q(z|x)$ and the prior $p(z) = \mathcal{N}(0, 1)$:

$$\mathcal{L}_{\text{KL}} = -\frac{1}{2}\sum_{j=1}^{d}\left(1 + \log(\sigma_{1,j}^2) - \mu_{1,j}^2 - \sigma_{1,j}^2\right)$$

with D and d representing the input and latent dimensions, respectively.

In an operational anomaly detection setting, the decoder's negative log-likelihood (often referred to as reconstruction likelihood) would serve as the anomaly score, with higher values indicating more anomalous inputs. However, since no ecological anomalies occurred during our three-month observation window, this pilot study evaluates baseline modeling rather than anomaly detection accuracy. Specifically, we report reconstruction errors: mean squared error (MSE) and mean absolute error (MAE) for acoustic indices, and overall mean absolute error (averaged across all pixels in each patch) for NDVI and NDMI patches, computed only on non-cloudy patches after SCL masking.

3.4 Experimental Setup

The general pipeline of the BAT system is shown in Figure 3. It consists of independent audio and visual pipelines designed to operate separately but eventually integrate into a unified decision-support framework.

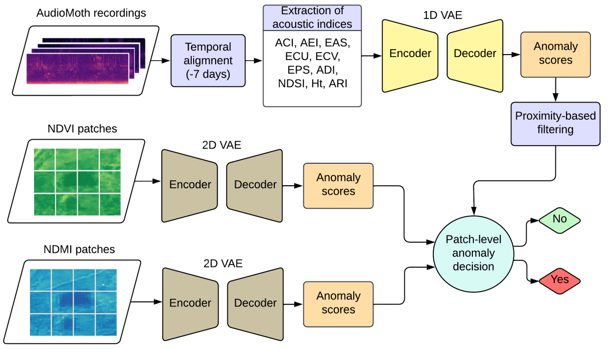

AudioMoth recordings
Temporal alignment (-7 days)
Extraction of acoustic indices
ACI, AEI, EAS, ECU, ECV, EPS, ADI, NDSI, Ht, ARI
1D VAE
Encoder
Decoder
Anomaly scores
Proximity-based filtering
NDVI patches
2D VAE
Encoder
Decoder
Anomaly scores
NDMI patches
2D VAE
Encoder
Decoder
Anomaly scores
Patch-level anomaly decision
No
Yes

Figure 3: The general pipeline of the BAT system.

In a full anomaly detection setting, the pipelines would use reconstruction likelihoods as anomaly scores and combine them across modalities. In this pilot, since no anomalies occurred, we only assess baseline modeling by training and evaluating the acoustic and satellite VAEs independently, reporting reconstruction errors as indicators of model performance.

3.4.1 *Audio Pipeline.*

The audio VAE uses a 10-dimensional input, with an encoder and decoder each containing one hidden layer of size 8 and ReLU activation. The latent space has dimension 4. The decoder outputs the reconstructed mean and log-variance of size 10.

Model evaluation used 5-fold cross-validation with folds defined by spatially clustered AudioMoth devices (~ 850 m minimum separation) to reduce data leakage. Models were trained for 30 epochs with a batch size of 512 using the Adam optimizer and a one-cycle learning rate schedule.

3.4.2 *Visual Pipeline.*

The satellite VAE takes a 16×16 pixel input (NDVI or NDMI) and uses three convolutional layers (32, 64, 128 filters) with ReLU activation in the encoder. The output is flattened and mapped to a latent space of dimension 4. The decoder upsamples using three transposed convolutional layers with ReLU, reconstructing the mean and log-variance patches of size 16×16.

Separate VAE models were trained for NDVI and NDMI using an 80/20 train-test split. Each model was trained for 20 epochs with a batch size of 32 using the Adam optimizer. The loss was computed only over non-cloudy pixels.

4 Results and Discussion

To examine temporal patterns, all indices were plotted over the study period as seen in Figure 4. Acoustic indices were averaged between 9AM and 3PM across all 10 AudioMoth devices to avoid nighttime inactivity and minimize dawn/dusk transitions. A 10-day smoothing window was applied to reduce day/night fluctuations. The indices remained relatively stable long-term, showing little trend and suggesting no major ecological disruptions and reflecting the stability of the forest soundscape over the study period.

Visual indices were averaged across all patches for each date. Both indices exhibit a gradual increase from early April to late June, consistent with seasonal greening. NDVI shows a smooth and consistent rise, indicating widespread vegetation growth. NDMI, while generally increasing, displays more irregular variation, particularly early in the season, likely reflecting transient moisture conditions. NDVI primarily tracks canopy structure and greenness, while NDMI is more sensitive to vegetation and soil moisture.

The audio pipeline VAE was evaluated using reconstruction MSE and MAE. Since all indices were normalized to the [0,1] range, errors are directly comparable. As shown in Figure 5, reconstruction errors are generally low, indicating that the model effectively captures the underlying structure of the acoustic data.

EPS and Ht showed the highest reconstruction error variability. This suggests they are more difficult to model but may provide sensitive signals of ecological change in future anomaly detection settings. Indices with consistently low reconstruction errors, on the other hand, indicate stable features that can serve as robust components of ecological baselines. These patterns highlight differences in how well various indices represent typical acoustic dynamics, which is central to establishing reliable baseline models.

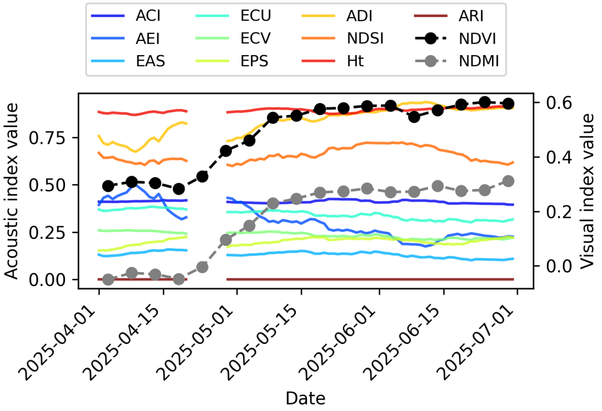


Figure 4: Index values over the study period.

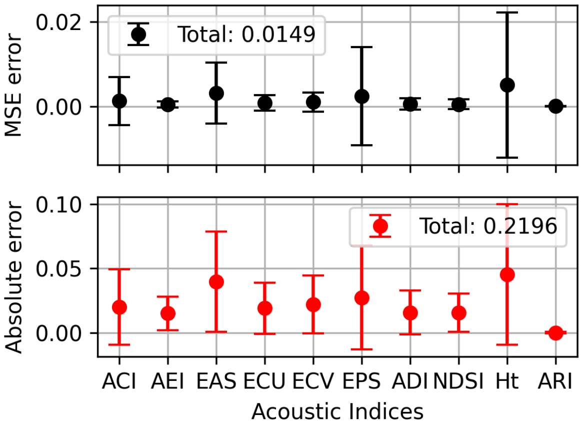


Figure 5: Reconstruction errors for acoustic indices.

The visual pipeline VAEs were evaluated using overall MAE per patch. As expected, errors were fairly uniform across pixels, indicating that the models reconstruct spatial patterns consistently without localized distortions. The average patch-level MAE (average across all $16 \times 16 = 256$ pixels across all images) was 7.17 ± 0.11 for NDVI and 9.65 ± 0.26 for NDMI. Given the [0, 1] normalization range of each pixel, the errors are relatively small and therefore reflect accurate reconstruction of vegetation and moisture dynamics.

The selected VAE models for both the acoustic and visual pipelines demonstrate strong reconstruction performance, with consistently low errors across acoustic indices and Sentinel-derived NDVI/NDMI patches. This confirms that the models effectively capture typical ecological patterns, which is the intended outcome of this pilot study. While further hyperparameter tuning could potentially reduce errors, the key result is that robust ecological baselines can be modeled. Anomaly detection itself will require expert-labeled events in future deployments, but these results provide the necessary technical foundation.

5 Conclusion

This work demonstrates the technical feasibility of using VAEs to model baseline ecological patterns from acoustic and satellite time series in a forested landscape. As a pilot study, it does not evaluate anomaly detection directly, since no anomalies occurred during the observation period. Instead, it establishes that robust models can be trained on available data, providing a foundation for future multimodal monitoring.

A critical next step is the collection of additional data over longer time frames and across multiple forest types, since actual ecological anomalies are rare and cannot be guaranteed within a short observation window. Detecting and validating anomalies will require expert labeling of such events once they occur. To this end, we are continuing data collection at Dyrehaven and planning expansions to other Danish forests (e.g., Thy, Amager, Lillebælt) to capture a wider range of ecological contexts and improve model generalization. Further development will also focus on refining acoustic preprocessing through time-window averaging or time-aware features and enhancing the visual pipeline with seasonal baselines, sequential models, and zone-specific approaches that account for spatial heterogeneity.

With expert input, longer-term recordings, and broader deployment, the BAT system can evolve from modeling site-specific baselines into a robust anomaly detection tool supporting scalable and long-term biodiversity monitoring.

Acknowledgements

This work was funded by Fauna Smart Technologies ApS under the European Space Agency (ESA) grant no. 4000147116, Biodiversity Assessment Tool.

References

[1] William R. L. Anderegg, Oriana S. Chegwidden, Grayson Badgley, Anna T. Trugman, Danny Cullenward, John T. Abatzoglou, Jeffrey A. Hicke, Jeremy Freeman, and Joseph J. Hamman. 2022. Future climate risks from stress, insects and fire across us forests. *Ecology Letters*, 25, 6, 1510–1520. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/ele.14018. DOI: https://doi.org/10.1111/ele.14018.

[2] Lucas P. Gaspar et al. 2023. Predicting bird diversity through acoustic indices within the atlantic forest biodiversity hotspot. *Frontiers in Remote Sensing*, 4, (Dec. 2023). DOI: 10.3389/frsen.2023.1283719.

[3] J.Wolfgang Wägele et al. 2022. Towards a multisensor station for automated biodiversity monitoring. *Basic and Applied Ecology*, 59, 105–138. DOI: https://doi.org/10.1016/j.baae.2022.01.003.

[4] Santiago Izquierdo-Tort, Andrea Alatorre, Paulina Arroyo-Gerala, Elizabeth Shapiro-Garza, Julia Naime, and Jérôme Dupras. 2024. Exploring local perceptions and drivers of engagement in biodiversity monitoring among participants in payments for ecosystem services schemes in southeastern mexico. *Conservation Biology*, 38, 6, e14282. eprint: https://conbio.onlinelibrary.wiley.com/doi/pdf/10.1111/cobi.14282. DOI: https://doi.org/10.1111/cobi.14282.

[5] Nathalie Pettorelli, Jake Williams, Henrike Schulte to Bühne, and Merry Crowson. 2025. Deep learning and satellite remote sensing for biodiversity monitoring and conservation. *Remote Sensing in Ecology and Conservation*, 11, 2, 123–132. eprint: https://zslpublications.onlinelibrary.wiley.com/doi/pdf/10.1002/rse2.415. DOI: https://doi.org/10.1002/rse2.415.

[6] Rory Gibb, Ella Browning, Paul Glover-Kapfer, and Kate E. Jones. 2019. Emerging opportunities and challenges for passive acoustics in ecological assessment and monitoring. *Methods in Ecology and Evolution*, 10, 2, 169–185. eprint: https://besjournals.onlinelibrary.wiley.com/doi/pdf/10.1111/2041-210X.13101. DOI: https://doi.org/10.1111/2041-210X.13101.

[7] D.A. Nieto-Mora, Susana Rodríguez-Buritica, Paula Rodríguez-Marín, J.D. Martínez-Vargaz, and Claudia Isaza-Narváez. 2023. Systematic review of machine learning methods applied to ecoacoustics and soundscape monitoring. *Heliyon*, 9, 10, e20275. DOI: https://doi.org/10.1016/j.heliyon.2023.e20275.

[8] Nathalie Pettorelli et al. 2018. Satellite remote sensing of ecosystem functions: opportunities, challenges and way forward. *Remote Sensing in Ecology and Conservation*, 4, 2, 71–93. eprint: https://zslpublications.onlinelibrary.wiley.com/doi/pdf/10.1002/rse2.59. DOI: https://doi.org/10.1002/rse2.59.

[9] Copernicus Data Space Ecosystem. 2015. Sentinel-2. (2015). https://dataspace.copernicus.eu/explore-data/data-collections/sentinel-data/sentinel-2.

[10] Luis J. Villanueva-Rivera, Bryan C. Pijanowski, Jarrod Doucette, and Burak Pekin. 2011. A primer of acoustic analysis for landscape ecologists. *Landscape Ecology*, 26, 9, (July 2011), 1233–1246. DOI: 10.1007/s10980-011-9636-9.

Development of a Lightweight Model for Detecting Solitary-Bee Buzz Using Pruning and Quantization for Edge Deployment

Ryo Yagi
yagi-ryo143@g.ecc.u-tokyo.ac.jp
The University of Tokyo
Tokyo, Japan

David Susič
David.Susic@ijs.si
Jožef Stefan Institute
Ljubljana, Slovenia

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

Miha Finžgar
miha.finzgar@senso4s.com
Senso4s
Trzin, Slovenia

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

Abstract

Passive acoustic monitoring is increasingly applied in studies of pollinators, both for biodiversity assessment and for the conservation of endangered species. A major challenge is that continuous recording generates large volumes of audio data, making centralized processing impractical. Edge computing offers a promising alternative, provided that the models are optimized for resource constraints of edge devices while maintaining acceptable performance and efficiency. In this work, which is our initial study of the edge computing approach, we developed and evaluated compact classifiers for detecting buzzes of solitary bees, extending previous work on acoustic monitoring. We systematically apply pruning and quantization to multiple models, exploring a range of compression settings. Performance is assessed in terms of mean F1-score and on-disk size under both cross-validation and leave-one-location-out protocols. Results indicate that substantial reductions in model size can be achieved with a minimal loss of performance, and that the optimal trade-offs depend on the evaluation setting; for example, in cross-validation, a 25.2 MiB baseline reaches 96.2% F1, while a 0.062 MiB model attains 92.5%, achieving an approximately 400-fold reduction in size with less than a 4-percentage-point drop. By analyzing the Pareto front of F1 vs. model size trade-offs, we identify configurations that balance robustness and resource constraints. Our early findings demonstrate the feasibility of deploying edge-ready acoustic models for scalable pollinator monitoring.

Keywords

edge deployment, lightweight model, pruning, quantization, bees

1 Introduction

Bees are widely recognized as major pollinators - animal pollinators including honey bees contribute to yield in 75% of key crop species and an estimated 35% of global crop production [1]. This indicates the importance of pollinator monitoring and protection.

Passive acoustic monitoring (PAM) is a non-invasive approach that continuously records environmental sound with deployed microphones to monitor animal activity. Because it reduces manual surveys and can operate continuously across space and time—even at night and under inclement weather—it has gained attention as a cost-effective biodiversity monitoring technology. PAM has been widely adopted for multiple taxa such as birds and bats; in ornithology, for example, the deep-learning system BirdNET [2] is already used operationally to identify species from passively collected field recordings. PAM is also applied to bee behavior monitoring: in social bees (such as honeybees or bumblebees), microphones and accelerometers placed inside or outside hives enable non-invasive, continuous surveillance of queen presence, swarming cues, and robbing [3]. For solitary bees, recordings at the entrance of nesting boxes are used to detect buzzing and to characterize presence/absence and activity rhythms [4].

In acoustic approaches for bee state monitoring, machine learning has been widely used to automatically determine activity and behavioral states from audio recordings. Prior work includes both classical machine-learning pipelines and deep-learning methods. Classical approaches such as SVM, k-NN, and Random Forests have been shown to be practical and effective [5, 6]. Meanwhile, several studies suggest that CNN-based deep learning models achieve superior performance compared with traditional machine-learning methods [7, 8].

However, if all long-term, continuous PAM recordings are uploaded to the cloud, features such as mel spectrograms and MFCCs are extracted there, and then analyzed using machine learning or deep learning models, the resulting data volumes become extremely large, which, in a centralized cloud-only workflow, (i) inflates communication cost by requiring all long-duration audio to be uploaded [9], (ii) raises privacy concerns as incidental human speech can accumulate in the cloud [10], (iii) introduces round-trip latency for feature extraction and inference that impedes timely detection, and (iv) exposes scalability limits as storage and compute demands grow with multi-site, long-term deployments. To address these issues, we developed a high-accuracy, lightweight deep model designed for edge deployment, capable of on-device preprocessing and inference for recorded audio. Here, the term lightweight refers to memory (both RAM and storage), but in a broader view it also refers to CPU/GPU requirements, latency requirements, and even battery constraints, which is beyond the scope of this paper. In our intended operation, audio is processed on-device

Information Society 2025, Ljubljana, Slovenia

https://doi.org/10.70314/is.2025.skui.6788

and only the result is sent to the cloud, enabling multi-site, long-term monitoring with reduced storage cost and latency, while preserving privacy and power efficiency.

As a first step toward edge-based bee monitoring with PAM, we designed and evaluated a lightweight CNN specialized for solitary-bee buzz detection (binary classifier distinguishing between buzz and no-buzz). To compress the model, we applied compression techniques such as structured pruning and int8 post-training quantization when appropriate, and we quantified the size–accuracy trade-offs under edge-oriented constraints.

2 Methodology

2.1 Dataset

We used the dataset collected for the purpose of the study by Susič et al. [4]. This dataset comprises acoustic recordings from nesting boxes of solitary bees (predominantly Osmia spp.) collected through a citizen-science project carried out in the Bela Krajina region in the southeastern Slovenia. The recordings were gathered from March 15 to May 26, 2023, resulting in 62 long recordings across seven sites, with a mean duration of 6 ± 2.5 hours per recording. For the purpose of this study, three recordings in total were randomly selected from different locations.

The recordings were converted to mono-channel audio, segmented into 4 s windows with 2 s overlap, transformed into Mel spectrograms (128×128) configured to cover 50–1450 Hz, and standardized using the mean and standard deviation across the dataset. For labeling, two annotators inspected the spectrograms and assigned buzz=1 or no-buzz=0.

2.2 Neural Network Architecture

We addressed binary detection of solitary-bee buzzing from Mel spectrograms. With memory-constrained edge deployment, we evaluate four lighter CNNs compared to the ResNet-9 used in [4]. Specifically, we consider MobileNetV2 [11] and three custom lightweight architectures named BeeNet1, BeeNet2, and BeeNet3, that adopt a depthwise separable convolutional design similar to MobileNetV1 [12]. Model sizes and parameter counts are summarized in Table 1 and the architectural details of the BeeNet variants are provided in Table 2. In all architectures, each convolutional layer is followed by batch normalization, BatchNorm, and ReLU activation, whereas dw stands for depthwise convolution. For MobileNetV2, we use the standard backbone and adapt it to spectrograms by converting the first convolution to a 1-channel input and replacing the final linear layer with a 1280 → 2 classifier. All other layers remain identical to the original MobileNetV2.

While the ResNet-9 approach achieves an F1-score exceeding 95% under five-fold cross-validation on the dataset [4], its 25.2 MiB size renders its deployment on a memory-limited edge devices impractical. Accordingly, we designed and configured compact CNNs (MobileNetV2 and the BeeNet family) and, as detailed below, applied quantization and pruning to systematically evaluate the accuracy–model-size trade-off.

The aim of this study is to clarify accuracy as a function of model size and the effects of lightweighting techniques under strict model-size constraints assuming deployment on MCUs. Accordingly, we adopt a lightweight and relatively simple architecture, with the smallest model containing approximately 6k parameters.

Table 1: Parameter counts and model sizes of the models used in this study

	ResNet-9	Mobilenetv2	Beenet 1	Beenet 2	Beenet 3
Parameters (k)	6585.5	2225.9	50.2	17.6	6.4
Model size (MiB)	25.2	8.7	0.215	0.084	0.036

2.3 Model Compression Methods

Deploying deep neural networks on memory-constrained edge devices necessitates model compression. We examined two complementary techniques: quantization and pruning.

2.3.1 *Quantization.*

Quantization maps floating-point weights and activations to low-bit integers, thereby reducing model size and computation at inference. Here, we adopted post-training quantization (PTQ) and converted the trained network to int8 without additional training. We used the QNNPACK backend in PyTorch for ARM targets. To minimize both saturation (clipping) and rounding error under the 8-bit representation and mitigate accuracy degradation, we performed calibration with up to 300 batches of representative inputs to estimate the scale and zero-point.

2.3.2 *Pruning.*

Pruning reduces model complexity by deleting parameters deemed unimportant, thereby decreasing memory and compute complexity without retraining from scratch. Pruning can be categorized into structured and unstructured approaches. We adopted structured pruning to realize memory savings and speed-ups on commodity hardware, as unstructured sparsity typically requires specialized hardware or software support to translate sparsity into acceleration [13].

Our pruning pipeline followed Han et al. [14]: (1) train, (2) prune, and (3) retrain (fine-tune). For filter (i.e., output-channel) selection, we followed the idea of Li et al. [15], ranking convolutional filters by the L1 norm of their weights and removing those with the smallest scores. We implemented this using PyTorch's torch-pruning, configuring the MagnitudePruner with L1-based importance. The selection of filters pruned was performed globally across layers. The target was controlled by a pruning ratio p; under channel-wise pruning, the resulting parameter-reduction rate was approximately $1 - (1 - p)^2$ [16, 17].

2.4 Experimental Setup

2.4.1 *Model Performance Evaluation Metrics.*

Because we were dealing with a class-imbalanced dataset (more no-buzz than buzz), we used the F1-score as the primary metric. F1 is the harmonic mean of precision and recall, enabling balanced assessment under imbalance.

2.4.2 *Evaluation Protocols.*

We evaluated the buzz-detecting models using two protocols, following [4]: cross-validation (CV) and leave-one-location-out (LOLO). The first approach is a standard test in machine-learning studies whereas the second one shows how well the model generalizes to the data coming from a previously unseen location with potentially different background noise. For CV, annotated segments (4 s windows) were partitioned into five folds; models were trained on four folds and evaluated on the remaining fold, and we reported the mean F1 across folds. Stratification ensures balanced distributions of the buzz/no-buzz classes and the three locations. To mitigate temporal leakage, we performed a time-aware data split.

Table 2: The architectures of BeeNet1, BeeNet2, BeeNet3

BeeNet1			BeeNet2			BeeNet3		
Type / Stride	Filter Shape	Input Size	Type / Stride	Filter Shape	Input Size	Type / Stride	Filter Shape	Input Size
Conv / s1	3 × 3 × 1 × 32	128 × 128 × 1	Conv / s1	3 × 3 × 1 × 32	128 × 128 × 1	Conv / s1	3 × 3 × 1 × 32	128 × 128 × 1
MaxPool / s2	Pool 2 × 2	128 × 128 × 32	MaxPool / s2	Pool 2 × 2	128 × 128 × 32	MaxPool / s2	Pool 2 × 2	128 × 128 × 32
Conv dw / s1	3 × 3 × 32 dw	64 × 64 × 32	Conv dw / s1	3 × 3 × 32 dw	64 × 64 × 32	Conv dw / s1	3 × 3 × 32 dw	64 × 64 × 32
Conv / s1	1 × 1 × 32 × 32	64 × 64 × 32	Conv / s1	1 × 1 × 32 × 32	64 × 64 × 32	Conv / s1	1 × 1 × 32 × 32	64 × 64 × 32
MaxPool / s2	Pool 2 × 2	64 × 64 × 32	MaxPool / s2	Pool 2 × 2	64 × 64 × 32	MaxPool / s2	Pool 2 × 2	64 × 64 × 32
Conv dw / s1	3 × 3 × 32 dw	32 × 32 × 32	Conv dw / s1	3 × 3 × 32 dw	32 × 32 × 32	Conv dw / s1	3 × 3 × 32 dw	32 × 32 × 32
Conv / s1	1 × 1 × 32 × 64	32 × 32 × 32	Conv / s1	1 × 1 × 32 × 64	32 × 32 × 32	Conv / s1	1 × 1 × 32 × 64	32 × 32 × 32
MaxPool / s2	Pool 2 × 2	32 × 32 × 64	MaxPool / s2	Pool 2 × 2	32 × 32 × 64	MaxPool / s8	Pool 8 × 8	32 × 32 × 64
Conv dw / s1	3 × 3 × 64 dw	16 × 16 × 64	Conv dw / s1	3 × 3 × 64 dw	16 × 16 × 64	FC / s1	1024 × 2	4 × 4 × 64
Conv / s1	1 × 1 × 64 × 128	16 × 16 × 64	Conv / s1	1 × 1 × 64 × 128	16 × 16 × 64	Softmax / s1	Classifier	1 × 1 × 2
MaxPool / s2	Pool 2 × 2	16 × 16 × 128	MaxPool / s4	Pool 4 × 4	16 × 16 × 128			
Conv dw / s1	3 × 3 × 128 dw	8 × 8 × 128	FC / s1	2048 × 2	4 × 4 × 128			
Conv / s1	1 × 1 × 128 × 256	8 × 8 × 128	Softmax / s1	Classifier	1 × 1 × 2			
MaxPool / s4	Pool 4 × 4	8 × 8 × 256						
FC / s1	1024 × 2	2 × 2 × 256						
Softmax / s1	Classifier	1 × 1 × 2						

LOLO assessed generalization across sites: models were trained on data from two of the three locations and evaluated on the held-out location, reporting the mean F1 across the three possible holds.

2.4.3 Hyperparameters. We trained the models with cross-entropy loss and the Adam optimizer, using a 1-cycle learning-rate schedule (maximum LR = 0.001), gradient clipping at 0.1, batch size 64, and 20 epochs. Compared to [4], the only change was increasing the number of epochs from 10 to 20. For pruning fine-tuning, we trained for 10 epochs with a fixed learning rate of 0.0001 and no scheduler. We compared the pruning ratios p of 0% (no pruning), 20%, 30%, and 50%.

3 Results

3.1 F1 vs. Model Size

For each model, we trained and evaluated a variety of combinations of pruning ratios and quantizations. Table 3 reports the mean F1 and on-disk model size (in MiB) for each setting. Figure 1 shows the plot of all configurations in the F1 − model-size plane for CV and LOLO, respectively, with the global Pareto front indicating the best trade-offs between model performance and its size denoted by a dashed line.

Even under tight memory budgets (< 100 KiB), competitive accuracy is achievable. For example, BeeNet1 (int8, p=0) attains 0.062MiB with CV F1 of 92.5% and LOLO F1 of 85.7%. Relative to ResNet-9 (float32, p=0), this represents an ∼ 400× reduction in model size while keeping F1 within 4 percentage points in both protocols, which is really promising for future edge deployment.

Performance degradation from int8 quantization is small: across many settings the F1 drop is about 1 percentage point (pp). With pruning, larger models exhibit smaller accuracy losses as p increases. For example, at p=50% ResNet-9 (float32) decreases only from 96.2% to 95.1% in CV and from 89.5% to 87.6% in LOLO, a decline of ≈ 2 percentage points in total. By contrast, the more compact BeeNet family is more sensitive: accuracy degrades markedly with p, and at p=50% most configurations lose ≥ 4 pp.

Inspection of the global Pareto front shows that many frontier points correspond to unpruned float32 or int8 models. At a fixed memory budget, lightly pruned or unpruned lightweight architectures often achieve higher accuracy than heavily pruned larger networks, indicating that purpose-built small models are preferable to aggressive pruning under the same size constraint.

A note on MobileNetV2 at p=30%: the trained model degenerated to predicting *no-buzz* for almost all inputs. This behavior may stem from a strong structured reduction under class imbalance and warrants further investigation.

4 Conclusions

We addressed buzz detection in acoustic recordings from solitary-bee nesting boxes, aiming to develop deep-learning models suitable for memory-constrained edge deployment. We designed or selected five CNN architectures and systematically measured the performance vs. model-size trade-offs under quantization and structured pruning. As a result, we obtained sub-100 KiB models achieving F1 scores of at least 92% in CV and 85% in LOLO experiments, indicating the feasibility and strong potential of accurate on-device inference.

For future work, we plan to train the models on additional datasets that we have collected to improve robustness and to deploy the models on real edge devices. Because our compression pipeline relied on simple techniques, we anticipate further gains by adopting a broader set of compression methods, such as knowledge distillation [18], quantization-aware training (QAT) [19], and neural architecture search (NAS) [20] to optimize model architectures under memory constraints, including number and sizes of the filters.

Acknowledgements

The authors acknowledge the Slovenian Research and Innovation Agency, grants PR-10495 and J7-50040, and Basic core funding P2-0209.

References

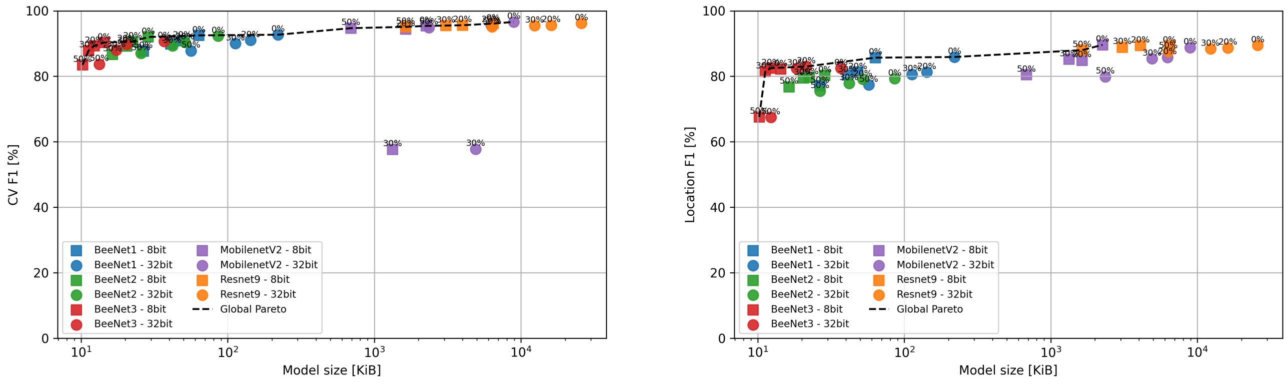
[1] Tom D Breeze, Alison P Bailey, Kelvin G Balcombe, and Simon G Potts. 2011. Pollination services in the uk: how important are honeybees? *Agriculture, Ecosystems & Environment*, 142, 3-4, 137–143.

[2] Stefan Kahl, Connor M. Wood, Maximilian Eibl, and Holger Klinck. 2021. Birdnet: a deep learning solution for avian diversity monitoring. *Ecological Informatics*, 61, 101236. doi: https://doi.org/10.1016/j.ecoinf.2021.101236.

[3] Mahsa Abdollahi, Pierre Giovenazzo, and Tiago H Falk. 2022. Automated beehive acoustics monitoring: a comprehensive review of the literature and recommendations for future work. *Applied Sciences*, 12, 8, 3920.

Table 3: Comparison of F1 scores and model sizes with quantization and pruning applied for five CNNs (ResNet-9, MobileNetV2, and BeeNet1/2/3)

Model (Quant.)	CV								LOLO							
	Pruning ratio (%)								*Pruning ratio (%)*							
	0		20		30		50		0		20		30		50	
	F1 (%)	Size (MiB)	F1 (%)	Size (MiB)	F1 (%)	Size (MiB)	F1 (%)	Size (MiB)	F1 (%)	Size (MiB)	F1 (%)	Size (MiB)	F1 (%)	Size (MiB)	F1 (%)	Size (MiB)
ResNet-9 (float32)	96.2	25.2	95.6	15.7	95.5	12.1	95.1	6.2	89.5	25.2	88.7	15.8	88.4	12.0	87.6	6.2
ResNet-9 (int8)	96.1	6.3	95.6	3.9	95.5	3.0	95.1	1.6	89.4	6.3	89.4	4.0	88.9	3.0	88.0	1.6
MobileNetV2 (float32)	96.6	8.7	95.6	6.1	57.8	4.8	94.9	2.3	88.8	8.7	85.8	6.1	85.4	4.8	79.9	2.3
MobileNetV2 (int8)	95.4	2.2	94.4	1.6	57.7	1.3	94.7	0.677	89.6	2.2	84.9	1.6	85.2	1.3	80.5	0.665
BeeNet1 (float32)	92.7	0.215	91.0	0.140	90.0	0.110	87.8	0.055	85.9	0.215	81.3	0.139	80.6	0.110	77.4	0.056
BeeNet1 (int8)	92.5	0.062	90.9	0.048	89.9	0.040	87.7	0.026	85.7	0.062	81.3	0.047	80.4	0.040	77.1	0.026
BeeNet2 (float32)	92.3	0.084	90.4	0.050	89.3	0.041	87.0	0.025	79.3	0.084	79.1	0.051	77.9	0.041	75.5	0.026
BeeNet2 (int8)	92.0	0.028	90.6	0.022	89.2	0.020	86.7	0.016	80.3	0.028	79.7	0.022	79.5	0.020	76.8	0.016
BeeNet3 (float32)	90.7	0.036	89.8	0.020	88.0	0.017	83.7	0.013	82.5	0.036	83.0	0.021	82.4	0.018	67.5	0.012
BeeNet3 (int8)	90.3	0.014	89.3	0.012	87.9	0.011	83.5	0.010	82.2	0.014	82.5	0.012	81.5	0.011	67.6	0.010



(a) Cross-validation (CV)

(b) Leave-one-location-out (LOLO)

Figure 1: F1–model-size trade-offs with the global Pareto frontier under CV and LOLO

[4] David Susič, Johanna A. Robinson, Danilo Bevk, and Anton Gradišek. 2025. Acoustic monitoring of solitary bee activity at nesting boxes. *Ecological Solutions and Evidence*, 6, 3, e70080. DOI: https://doi.org/10.1002/2688-8319.70080.

[5] Alison Pereira Ribeiro, Nádia Felix Felipe da Silva, Fernanda Neiva Mesquita, Priscila de Cássia Souza Araújo, Thierson Couto Rosa, and José Neiva Mesquita-Neto. 2021. Machine learning approach for automatic recognition of tomato-pollinating bees based on their buzzing-sounds. *PLOS Computational Biology*, 17, 9, (Sept. 2021), 1–21. DOI: 10.1371/journal.pcbi.1009426.

[6] Antonio Robles-Guerrero, Tonatiuh Saucedo-Anaya, Carlos A. Guerrero-Mendez, Salvador Gómez-Jiménez, and David J. Navarro-Solís. 2023. Comparative study of machine learning models for bee colony acoustic pattern classification on low computational resources. *Sensors*, 23, 1. DOI: 10.3390/s23010460.

[7] Jaehoon Kim, Jeongkyu Oh, and Tae-Young Heo. 2021. Acoustic scene classification and visualization of beehive sounds using machine learning algorithms and grad-cam. *Mathematical Problems in Engineering*, 2021, 1, 5594498.

[8] Vladimir Kulyukin, Sarbajit Mukherjee, and Prakhar Amlathe. 2018. Toward audio beehive monitoring: deep learning vs. standard machine learning in classifying beehive audio samples. *Applied Sciences*, 8, 9, 1573.

[9] Carrie C Wall, Samara M Haver, Leila T Hatch, Jennifer Miksis-Olds, Rob Bochenek, Robert P Dziak, and Jason Gedamke. 2021. The next wave of passive acoustic data management: how centralized access can enhance science. *Frontiers in Marine Science*, 8, 703682.

[10] Benjamin Cretois, Carolyn M Rosten, and Sarab S Sethi. 2022. Voice activity detection in eco-acoustic data enables privacy protection and is a proxy for human disturbance. *Methods in Ecology and Evolution*, 13, 12, 2865–2874.

[11] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. Mobilenetv2: inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4510–4520.

[12] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. 2017. Mobilenets: efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

[13] Hongrong Cheng, Miao Zhang, and Javen Qinfeng Shi. 2024. A survey on deep neural network pruning: taxonomy, comparison, analysis, and recommendations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46, 12, 10558–10578. DOI: 10.1109/TPAMI.2024.3447085.

[14] Song Han, Jeff Pool, John Tran, and William Dally. 2015. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28.

[15] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. 2016. Pruning filters for efficient convnets. *arXiv preprint arXiv:1608.08710*.

[16] Gongfan Fang, Xinyin Ma, Mingli Song, Michael Bi Mi, and Xinchao Wang. 2023. Depgraph: towards any structural pruning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[17] Gongfan Fang and contributors. 2023. Torch-pruning: structural pruning for pytorch. (2023). https://github.com/VainF/Torch-Pruning.

[18] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

[19] Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. 2018. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2704–2713.

[20] Barret Zoph and Quoc V. Le. 2017. Neural architecture search with reinforcement learning. (2017). https://arxiv.org/abs/1611.01578 arXiv: 1611.01578 [cs.LG].

Interpretable Predictive Clustering Tree for Post-Intubation Hypotension Assessment

Estefanía Žugelj Tapia
Institute of Physiology
University of Ljubljana, Medical faculty
Ljubljana, Slovenia
estefania.tapia@mf.uni-lj.si

Borut Kirn
Institute of Physiology
University of Ljubljana, Medical faculty
Ljubljana, Slovenia
borut.kirn@mf.uni-lj.si

Sašo Džeroski
Department of Knowledge Technologies
Jožef Stefan Institute
Ljubljana, Slovenia
saso.dzeroski@ijs.si

Abstract

Intraoperative hypotension following intubation is a clinically significant event associated with increased morbidity and mortality. This study presents an interpretable predictive clustering tree (PCT) model designed for multi-target prediction of hypotensive outcomes, including the prediction of minimum and maximum mean arterial pressure (MAP) values during hypotension in the post-induction period. The multi-target regression trees (MTRT) were evaluated using 10-fold cross-validation, and feature importance was assessed via a random forest model. Compared to the original tree, the pruned model demonstrated improved generalization and reduced complexity, with fewer nodes and enhanced interpretability. The pruned tree structure enabled clear decision thresholds based on modifiable variables such as MAP_after_5min, MAP_basal, and Propofol dose. While the random forest achieved the highest performance and had high complexity, its feature importance ranking analysis supported the relevance of the attributes retained in the pruned model and provided complementary insights, highlighting globally relevant features, such as SBP_after_5min, that were not prioritized in the single trees. These findings support the use of interpretable models in clinical decision-support to anticipate and potentially modify the occurrence of post-intubation hypotension.

Keywords

multi-target prediction, interpretable machine learning, decision tree pruning, feature importance, post-intubation, intraoperative hypotension

1 Introduction

Intubation is a common procedure in emergency departments and operating rooms, typically performed immediately after the administration of induction agents. These agents have been associated with hemodynamic instability and post-induction hypotension (PIH), frequently defined as mean arterial pressure (MAP) <65 mmHg[1]. Particularly, in perioperative medicine, PIH has been related to worse postoperative outcomes, increased comorbidity, and mortality[2,3]. PIH occurrence is limited to the first 30 minutes post-induction, as this period is directly affected by anesthesia effects, and is usually not related to complex factors due to surgery[4]. Regarding the risk factors, a post hoc analysis in a surgical population of patients at risk of aspiration of gastric content identified different risk factors associated with it in the multivariate analysis: age, a higher baseline heart rate, bowel occlusion requiring nasogastric tube placement before intubation, and the use of remifentanil. A prospective multicenter study found that in the group with hypotension, the dose (mg/kg) of Propofol was significantly higher at 5 and 10 minutes after induction[5]. On the other hand, the following protective factors have been described: low doses of ketamine and basal systolic blood pressure (SBP)[2].

Previous studies have employed traditional multivariate analysis to identify risk factors and have focused on predicting a single target: the presence of hypotension[2,4,5]. However, predicting multiple outcomes simultaneously can capture complex interactions and provide more informative insights, aiding clinical decision-making and support. Therefore, the hypothesis of this study is that predicting multiple outcomes of PIH simultaneously can effectively identify which variables are most influential in predicting PIH. Overall, this study contributes to the prediction of PIH, which can help anesthesiologists to make better decisions during induction, potentially improving patient outcomes.

2 Methods

Predictive clustering trees (PCT) are a machine learning framework that unifies clustering and prediction tasks. In this framework, the node at the roof (the top node or root node) corresponds to the cluster that contains all the data, and each subsequent split partitions the data to minimize intra-cluster variance. CLUS is a free software that implements this framework and supports multi-target prediction. In a multi-target regression tree (MTRT), the obtained tree is more reliable in explaining the dependencies between variables, and the prediction is a vector of values of the target attributes[6,7]. For this reason, CLUS version 2.12.8 was chosen as the software for this retrospective analysis. The documentation and latest version can be found at: https://github.com/knowledge-technologies/clus/tree/main.

Data was sourced from the subset SIS of the MOVER database (https://mover.ics.uci.edu/) —a public database of anonymized patients undergoing various types of surgery[8].

The inclusion and exclusion criteria were the following:

Information Society 2025, 6–10 October 2025, Ljubljana, Slovenia
© 2025 Copyright held by the owner/author(s).
http://doi.org/DOI 10.70314/is.2025.skui.9188

- *Inclusion criteria:* 1) patients who underwent major surgery procedures with documented application and dose of one of the next medications during induction of general anesthesia: 'Midazolam', 'Propofol', 'Fentanyl', 'Succinylcholine', 'Ketamine', 'Cisatracurium', 'Etomidate', 'Vecuronium', and 'Rocuronium', 2) high temporal resolution vital signs of systolic blood pressure (SBP), diastolic blood pressure (DBP), and mean arterial pressure (MAP) measured from the radial arterial line, registered before the time of intubation and 30 minutes after it, with at least one measure of MAP < 65mmHg during the post-intubation period.

-*Exclusion criteria:* Patients with vital signs out of physiological ranges (MAP <30mmHg and MAP > 200mmHg), and patients who do not meet the inclusion criteria.

As descriptive and target attributes of the learning problem (see Table 1), the following variables were calculated: 1) MAP_basal: average of MAP measures before intubation, 2) SBP_basal: average of SBP measures before intubation, 3) DBP_basal: average of DBP measures before intubation, 4) MAP_after_5min: average of MAP measurements taken after intubation, over a 5-minute period, 5) SBP_after_5min: average of SBP measurements taken after intubation, over a 5-minute period, 6) DBP_after_5min: average of DBP measurements taken after intubation, over a 5-minute period, 7) Min_MAP<65: Minimum MAP <65 mmHg registered from the intubation up to 30 minutes after, 8) Max_MAP<65: Maximum MAP <65 mmHg registered from the intubation up to 30 minutes after, 9) MAP<65_count: Counts of registered measurements <65mmHg over the 30 minutes interval after intubation, 10) MAP_mean_after_30min: average of MAP measures over 30 minutes interval after intubation, 11) SBP_mean_after_30min: average of SBP measures over 30 minutes interval after intubation, 12) MAP<65_mean_after_30_min: average of MAP measures <65 mmHg over 30 minutes interval after intubation, and 13) Body mass index (BMI): weight / ((height / 100)^2).

During data preparation, missing values of the height attribute were replaced with the mean value of the attribute.

Table 1: Descriptive and target attributes

Descriptive attributes (20)	Target attributes (6)
MAP_basal	Min_MAP<65
SBP_basal	Max_MAP<65
DBP_basal	MAP<65_count
MAP_after_5min	MAP<65_mean_after_30_min
SBP_after_5min	MAP_mean_after_30min
DBP_after_5min	SBP_mean_after_30min
Age	
Gender	
Height	
Weight	
BMI	
Midazolam (cumulative dose)	
Propofol (cum. dose)	
Fentanyl (cum. dose)	
Succinylcholine (cum. dose)	
Ketamine (cum. dose)	
Cisatracurium (cum. dose)	
Etomidate (cum. dose)	
Vecuronium (cum. dose)	
Rocuronium (cum. dose)	

After defining the descriptive and target attributes, the entire dataset of 340 patients was split into training and test sets using the sklearn library and the train_test_split function: 80% of the dataset was used for training (272 patients) and 20% for testing (68 patients). To run CLUS, the training and test sets were converted to ARFF format. Corresponding settings file (.s) were created to define the model parameters for the MTRT tasks. Both single-tree and ensemble models were trained, as summarized in Table 2.

Table 2: Tree and ensemble specifications for each respective MTRT.

Model	Predictive clustering tree (PCT)	Random forest
Heuristic	Variance Reduction	Variance Reduction
Pruning Method	M5Multi	-
Ensemble Method	-	RForest
Feature Ranking	-	Genie3

As an alternative to the train/test split, when running CLUS, the -xval command-line option was used to perform cross-validation on all 340 examples. The number of folds (n = 10) was previously specified in the settings file.

Model performance was evaluated using the following metrics: Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Relative Squared Error (RRMSE), and Pearson correlation coefficient (r^2), computed on both training and test sets.

3 Results

After applying the exclusion criteria, we were left with 340 patients. Figure 1 illustrates the flow chart for patient selection, and Table 3 shows their demographic characteristics.

Table 3: Data set population characteristics

Age, years, mean (SD)	58.9 (18.9)
Gender (male), count	201
Weight, kg, mean (SD)	78.6 (23.1)
Height, cm, mean (SD)	168.4 (11.1)
BMI, kg/m2, mean (SD)	1.5 (6.8)

3.1 Complexity of the Models and Structure

The induction time for the pruned model was significantly shorter (0.032 seconds) compared to the original model (1.622 seconds), reflecting its reduced complexity. Structurally, the original tree consisted of 241 nodes, 121 leaves, and a depth of 17, whereas the pruned tree was noticeably simpler, with only 19 nodes, 10 leaves, and a depth of 6.

Additionally, the ensemble random forest model, composed of 100 trees, contained a total of 21,050 nodes and 10,575 leaves, with an average tree depth of 154, indicating a significantly higher complexity and capacity for capturing intricate patterns in the data.

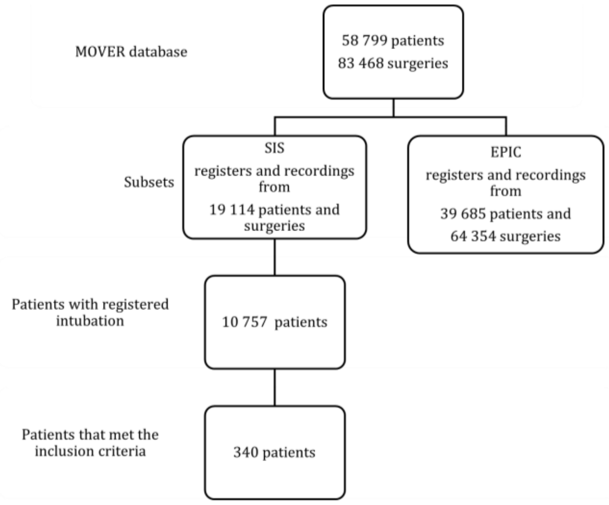

MOVER database
58 799 patients
83 468 surgeries
Subsets
SIS
registers and recordings from
19 114 patients and surgeries
EPIC
registers and recordings from
39 685 patients and
64 354 surgeries
Patients with registered intubation
10 757 patients
Patients that met the inclusion criteria
340 patients

Figure 1: Overview of sample population included in this study.

3.2 Model Performance

The forest with 100 trees exhibits the best performance overall. However, pruning significantly simplified the original model while retaining, and even improving, its predictive power, with lower testing errors for MAE, MSE, RMSE, and RRMSE compared to the original tree (See Table 4).

Table 4: Metrics for training and testing errors (Train/Test)

Metric	Default	Original (Unpruned)	Pruned	Forest (100 trees)
MAE	7.27 / 7.30	2.58 / 7.55	5.41 / 6.18	2.93 / 5.22
MSE	109.35 / 110.12	17.77 / 120.54	61.39 / 83.03	18.41 / 51.05
RMSE	9.41 / 9.45	3.81 / 10.15	7.09 / 8.33	3.96 / 6.76
RRMSE	1.00 / 1.00	0.42 / 1.13	0.76 / 0.91	0.44 / 0.86
Pearson r^2	– / 0.04	0.82 / 0.14	0.42 / 0.21	0.89 / 0.26

3.3 Cross-Validation Results

The 10-fold cross-validation was conducted using all 340 examples, with an induction time of 0.26 sec for the single tree and of 9.747 sec for the ensemble random forest. The mean number of tests for the original model was 267, for the pruned model 39.2, and for the random forest 100.

As shown in Table 5, the absolute error metrics (MAE, MSE, RMSE) were higher when using a train/test split, however the cross-validation approach yielded lower testing errors for RRMSE and higher Pearson r^2 values.

Table 5: Cross-validation metrics for training and testing errors (Train / Test)

Metric	Default	Original (Unpruned)	Pruned	Forest (100 trees)
MAE	13.62 / 13.69	1.80 / 10.49	5.78 / 9.28	2.82 / 5.6
MSE	300.15 / 302.3	9.22 / 193.3	64.2 / 150.5	16.83 / 63.15
RMSE	17.32 / 17.39	3.04 / 13.90	8.01 / 12.27	3.81 / 7.41
RRMSE	1.00 / 1.00	0.18 / 0.80	0.46 / 0.70	0.43 / 0.84
Pearson r^2	0.0003 / 0.02	0.97 / 0.45	0.79 / 0.52	0.89 / 0.28

Note that cross-validation yields more realistic estimates of error on unseen examples as compared to a single train-test split.

3.4 Original Model

As stated in section 3.1, the original model contains 241 nodes and 121 leaves. MAP_after_5min is at the root node, followed by MAP_basal, these two variables repeat along the tree on more than one occasion. Except for cisatracurium, ketamine, and etomidate, in the remaining nodes, the rest of the descriptive attributes appear at least once, showing different thresholds.

3.5 Pruned Model

In the pruned model, the descriptive attributes retained for multi-target prediction were MAP_after_5min, MAP_basal, BMI, SBP_basal, DBP_after_5min, and Propofol dose. Compared to the original tree, the pruned model demonstrated improved generalization and interpretability, with a significantly reduced number of nodes, as illustrated in Figure 2.

The highest predicted values for the target attributes—97.9 mmHg for MAP_mean_after_30min and 149.8 mmHg for SBP_mean_after_30min—were observed when MAP_after_5min exceeded 93 mmHg and SBP_basal was greater than 181 mmHg.

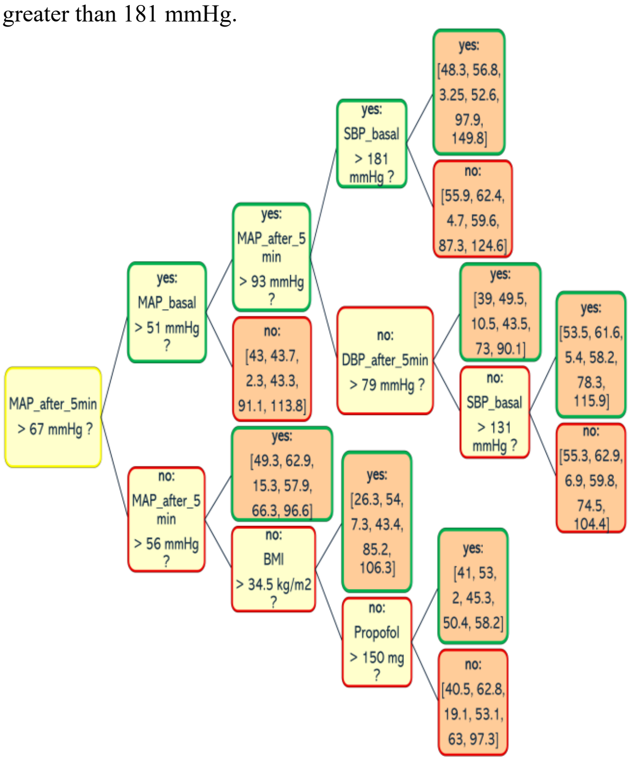

MAP_after_5min > 67 mmHg ?
yes: MAP_basal > 51 mmHg ?
yes: MAP_after_5 min > 93 mmHg ?
yes: SBP_basal > 181 mmHg ?
yes: [48.3, 56.8, 3.25, 52.6, 97.9, 149.8]
no: [55.9, 62.4, 4.7, 59.6, 87.3, 124.6]
no: DBP_after_5min > 79 mmHg ?
yes: [39, 49.5, 10.5, 43.5, 73, 90.1]
no: SBP_basal > 131 mmHg ?
yes: [53.5, 61.6, 5.4, 58.2, 78.3, 115.9]
no: [55.3, 62.9, 6.9, 59.8, 74.5, 104.4]
no: [43, 43.7, 2.3, 43.3, 91.1, 113.8]
no: MAP_after_5 min > 56 mmHg ?
yes: [49.3, 62.9, 15.3, 57.9, 66.3, 96.6]
no: BMI > 34.5 kg/m2 ?
yes: [26.3, 54, 7.3, 43.4, 85.2, 106.3]
no: Propofol > 150 mg ?
yes: [41, 53, 2, 45.3, 50.4, 58.2]
no: [40.5, 62.8, 19.1, 53.1, 63, 97.3]

Figure 2: Pruned tree, predicting min_MAP<65, max_MAP<65, MAP<65_count, MAP<65_mean_after_30_min, MAP_mean_after_30min, and SBP_mean_after_30min. Leaves display predictions in orange.

On the other hand, the lowest predicted values of these target variables—50.4 and 58.2 mmHg— were derived from the following nodes: MAP_after_5min below 56 mmHg, BMI < 34.5 kg/m^2, and the Propofol dose >150 mg. Additionally, the leaf node corresponding to BMI >34.5 kg/m^2 yielded the deepest value for min_MAP <65, at 26.3 mmHg.

Other notably low predictions related to hypotension included max_MAP<65 at 43.7 mmHg and MAP<65_mean_after_30_min at 43.3 mmHg, both derived from the node where MAP_basal was below 51 mmHg.

3.6 Forest and Feature Ranking

Despite the complexity of the forest with 100 trees, the feature ranking, where feature importance was assessed using the Genie3 score, helps to understand the descriptive attributes that mainly contributed to the final multi-target prediction. Figure 3 lists the first eleven descriptive attributes, ranked by their corresponding importance score. MAP_after_5min and SBP_after_5min are clearly the most influential features in the model; MAP_basal and SBP_basal also contribute significantly, closely following in importance.

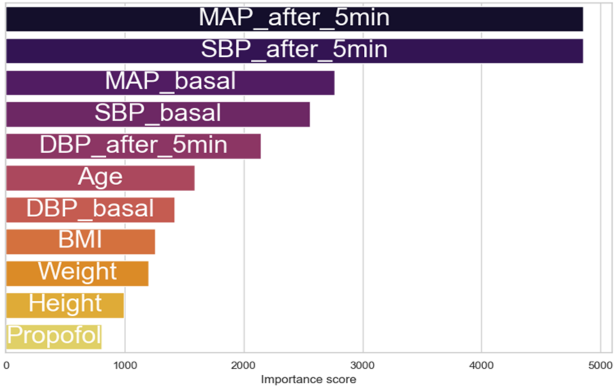


Figure 3: Descriptive attributes contributing most to the random forest's prediction, sorted by importance score.

4 Discussion & Conclusion

The advantages of using a predictive clustering method for multi-target prediction include the ability to capture complex interactions between descriptive attributes and the simultaneous prediction of multiple outcomes [6,7]. A key novelty of this study is its focus on predicting multiple outcomes related to hypotension. This multi-target approach provides a more comprehensive overview and enhances clinical decision-support. In clinical practice, anesthesiologists need to anticipate and often ask themselves: How low will MAP values drop? How will MAP evolve throughout the procedure? This is highly relevant because deeper and longer hypotensive episodes increase the presence of adverse events associated with intraoperative hypotension[3,4].

In this study, the pruned model included among the most important variables for the multi-target prediction MAP_after_5min and MAP_basal. Previous studies have significantly associated PIH with the basal or pre-induction MAP[2,4,5], and our results confirm this observation: In the node root, the MAP value was the most relevant when calculated immediately 5 minutes after intubation, specifically with a decisive threshold of 67 mmHg. To diminish the impact of the basal blood pressure values in the occurrence of PIH episodes, some proposals include discontinuing renin–angiotensin–aldosterone system antagonists the day of the surgery and proactive measures to elevate preoperative values to relieve the effect of the anesthetic medications, which could prevent the appearance of PIH [3,4].

The obtained pruned predictive clustering tree model showed lower testing errors across all metrics compared to the original tree, with improved performance, interpretability, and generalization. Nevertheless, the random forest model performed the best. Regardless of the complexity of the ensemble model, the feature ranking provided valuable insights into the contribution of each attribute to the final prediction; some of these top-ranked features also appear along the nodes of the unpruned and pruned trees. By aggregating importance across multiple trees, random forests can highlight globally relevant features that may not dominate early decision paths in a single tree. For example, SBP_after_5min was ranked second in importance, but it did not appear in the top splits of the unpruned tree. In the pruned tree, BMI and Propofol dose are included, but SBP_after_5min, age, and DBP_basal, which ranked higher than BMI and Propofol dose, are not incorporated in the pruned tree. The association between higher age and PIH has been noted in the past [2,5], and it is a variable usually considered during risk evaluation; however, it is not a modifiable attribute.

In sum, this study demonstrates that interpretable models, such as pruned trees, when supported by feature importance from high-performing models, can validate and offer clear, decisive thresholds of modifiable and actionable variables that impact MAP values in the post-induction period, thereby reducing PIH-related comorbidity and mortality. This highlights its potential utility as a decision support tool in clinical settings.

References

[1] Salmasi V, Maheshwari K, Yang D, Mascha EJ, Singh A, Sessler DI, et al. Relationship between Intraoperative Hypotension, Defined by Either Reduction from Baseline or Absolute Thresholds, and Acute Kidney and Myocardial Injury after Noncardiac Surgery. Anesthesiology 2017;126(1):47–65. DOI: https://doi.org/10.1097/ALN.0000000000001432

[2] Grillot N, Gonzalez V, Deransy R, Rouhani A, Cintrat G, Rooze P, et al. Post-induction hypotension during rapid sequence intubation in the operating room: A post hoc analysis of the randomized controlled REMICRUSH trial. Anaesth Crit Care Pain Med 2025; 44(3):101502. DOI: https://doi.org/10.1016/j.accpm.2025.101502

[3] Sessler DI, Bloomstone JA, Aronson S, Berry C, Gan TJ, Kellum JA, et al. Perioperative Quality Initiative consensus statement on intraoperative blood pressure, risk and outcomes for elective surgery. Br J Anaesth 2019;122(5):563–74. DOI: https://doi.org/10.1016/j.bja.2019.01.013

[4] Südfeld S, Brechnitz S, Wagner JY, Reese PC, Pinnschmidt HO, Reuter DA, et al. Post-induction hypotension and early intraoperative hypotension associated with general anaesthesia. Br J Anaesth 2017;119(1):57–64. DOI: https://doi.org/10.1093/bja/aex127

[5] Jor O, Maca J, Koutna J, Gemrotova M, Vymazal T, Litschmannova M, et al. Hypotension after induction of general anesthesia: occurrence, risk factors, and therapy. A prospective multicentre observational study. J Anesth 2018; 32(5):673–80. DOI: https://doi.org/10.1007/s00540-018-2532-6

[6] Kocev D, Vens C, Struyf J, Džeroski S. Tree ensembles for predicting structured outputs. Pattern Recognit 2012;46:817–33. DOI: https://doi.org/10.1016/j.patcog.2012.09.023

[7] Petković M, Levatić J, Kocev D, Breskvar M, Džeroski S. CLUSplus: A decision tree-based framework for predicting structured outputs. SoftwareX 2023; 24:101526. DOI: https://doi.org/10.1016/j.softx.2023.101526

[8] Samad M, Angel M, Rinehart J, Kanomata Y, Baldi P, Cannesson M. Medical Informatics Operating Room Vitals and Events Repository (MOVER): a public-access operating room database. JAMIA Open 2023;6(4). DOI: https://doi.org/10.1093/jamiaopen/ooad084

Indeks avtorjev / Author index