1st International Scientific and Practical Conference

«Computational Intelligence and Smart Systems»

Lviv, Ukraine
October 10-12

Міжнародна науково-практична конференція

«Обчислювальний інтелект і смарт-системи» CISS-2024

Politechnika Łódzka

FH CAMPUS WIEN
UNIVERSITY OF APPLIED SCIENCES

UNIVERZITA KOMENSKÉHO V BRATISLAVE

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Ministry of Education and Science of Ukraine
Lviv Polytechnic National University (Ukraine)
Comenius University Bratislava (Slovakia)
Lodz University of Technology (Poland)
University of Applied Sciences Campus Vienna (Austria)

Proceedings

of the 1st International Scientific and Practical Conference

«COMPUTATIONAL INTELLIGENCE AND SMART SYSTEMS» (CISS-2024)

10–12 October 2024
Lviv, Ukraine

Lviv
«ATB» Publishing
2024

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Університет Коменського в Братиславі (Словаччина)
Лодзька Політехніка (Польща)
Університет прикладних наук у Відні (Австрія)

Матеріали

1-ої Міжнародної науково-практичної конференції

«ОБЧИСЛЮВАЛЬНИЙ ІНТЕЛЕКТ І СМАРТ-СИСТЕМИ» (CISS-2024)

10–12 жовтня 2024
Львів, Україна

Львів
Видавництво ННВК «АТБ»
2024

UDC 004

Computational intelligence and smart systems: Proceedings of the 1st International Scientific and Practical Conference (CISS-2024), 10–12 October 2024. – Lviv Polytechnic National University. – Lviv: «ATB» Publishing, 2024.
ISBN 978-966-2042-89-4

The publication is intended for scientists, teachers, students, public organizations and all those interested in the use of computing intelligence and smart systems.

UDC 004

Responsible for the issue – Mariana Seneta and Viktor Zvarych

The materials are presented in the author's edition.

The authors are responsible for the content of the materials, their compliance with the requirements of the current spelling and the accuracy of facts and statistics.

ISBN 978-966-2042-89-4

УДК 004

Обчислювальний інтелект і смарт-системи: матеріали 1-ої Міжнародної науково-практичної конференції (CISS-2024), 10–12 жовтня 2024 р. – Національний університет «Львівська політехніка». – Львів: Видавництво ННВК «АТБ», 2024.
ISBN 978-966-2042-89-4

Видання призначене для науковців, викладачів, студентів, громадських організацій та усіх зацікавлених з питань використання обчислювального інтелекту і смарт-систем.

УДК 004

Відповідальні за випуск – Мар'яна СЕНЕТА та Віктор ЗВАРИЧ

Матеріали подано в авторській редакції.

ISBN 978-966-2042-89-4

PROGRAM COMMITTEE

Vasyl TESLYUK	**Chairman**, Lviv Polytechnic National University, Ukraine
Natalia KRYVINSKA	Comenius University Bratislava, Slovakia
Aneta PONISZEWSKA-MARANDA	Lodz University of Technology, Poland
Igor MILADINOVIC	University of Applied Sciences Campus Vienna, Austria
Mykola MEDYKOVSKYY	Lviv Polytechnic National University, Ukraine
Ivan TSMOTS	Lviv Polytechnic National University, Ukraine
Oleh BEREZSKY	Lviv Polytechnic National University, Ukraine
Roman TKACHENKO	Lviv Polytechnic National University, Ukraine
Ihor PROTSKO	Lviv Polytechnic National University, Ukraine
Volodymyr RIZNYK	Lviv Polytechnic National University, Ukraine
Liubomyr SIKORA	Lviv Polytechnic National University, Ukraine
Vasyl LYTVYN	Lviv Polytechnic National University, Ukraine
Volodymyr LYTVYNENKO	Kherson National Technical University, Ukraine
Iurii KRAK	Taras Shevchenko National University of Kyiv, Ukraine
Andrii SAFONYK	National University of Water and Environmental Engineering, Ukraine
Viktor SINEGLAZOV	National Aviation University, Ukraine
Victoria VYSOTSKA	Lviv Polytechnic National University, Ukraine
Olena PAVLIUK	Lviv Polytechnic National University, Ukraine
Iryna KAZYMYRA	Lviv Polytechnic National University, Ukraine
Yuriy OPOTYAK	Lviv Polytechnic National University, Ukraine
Kvitoslava OBELOVSKA	Lviv Polytechnic National University, Ukraine
Anastasiya DOROSHENKO	Lviv Polytechnic National University, Ukraine
Nataliia LYSA	Lviv Polytechnic National University, Ukraine
Ivan IZONIN	Lviv Polytechnic National University, Ukraine
Pavlo SAI	Polish Academy of Sciences, Warsaw, Poland
Ricardo J. RODRÍGUEZ	University of Zaragoza, Spain
Winfried AUZINGER	Institute of Analysis and Scientific Computing, Vienna, Austria
Tadeusz CZACHORSKI	Institute of Theoretical and Applied Informatics, Poland
Hanna TERLETSKA	Middle Tennessee State University, United States of America
Gordon HUNTER	Kingston University, London, United Kingdom
Jamshid DEHMESHKI	Kingston University, London, United Kingdom

ORGANIZING COMMITTEE

Ivan TSMOTS	**Chairman**, Lviv Polytechnic National University, Ukraine
Mariana SENETA	**Responsible Secretary**, Lviv Polytechnic National University, Ukraine
Vasyl DUBUK	Lviv Polytechnic National University, Ukraine
Nazariy RIPAK	Lviv Polytechnic National University, Ukraine
Artem KAZARIAN	Lviv Polytechnic National University, Ukraine
Volodymyr ANTONIV	Lviv Polytechnic National University, Ukraine
Viktor ZVARYCH	Lviv Polytechnic National University, Ukraine
Dmytro OSTROVKA	Lviv Polytechnic National University, Ukraine
Yurii LUKASHCHUK	Lviv Polytechnic National University, Ukraine
Roman SYDORENKO	Lviv Polytechnic National University, Ukraine
Kateryna TKACHUK	Lviv Polytechnic National University, Ukraine
Oleksandr VERNYHOR	Lviv Polytechnic National University, Ukraine

PREFACE

Proceedings contain the materials of reports presented at the 1st International Scientific and Practical Conference on Computational Intelligence and Smart Systems (CISS-2024), which was organized by the Department of Automated Control Systems of the Institute of Computer Sciences and Information Technologies (Lviv Polytechnic National University). The Conference co-organizers were Comenius University Bratislava (Slovakia), Lodz University of Technology (Poland), and University of Applied Sciences Campus Vienna (Austria).

The Conference is dedicated to the 50th anniversary of the founding of the Department of Automated Control Systems at Lviv Polytechnic National University.

The topic of the scientific meeting is actual and covers research from a wide range of fields of science and technology.

Thematic directions of the Conference:

- Methods and means of computational intelligence
- Neuro-fuzzy methods of controlling robotics systems
- Augmented and virtual reality in smart systems
- Methods of computational intelligence in medicine
- Methods of computational intelligence in smart systems
- Methods and means of managing a smart enterprise, a smart home, a smart city
- Security of smart networks

The topics of particular Conference sections are related to the Erasmus+ educational projects of the Jean Monnet Module, which are implemented at the Department of Automated Control Systems (AR4EDU, TrustAI and DataProEU).

The section on reviewing developments and own projects in computational intelligence and smart systems was of particular interest.

During the Conference, it was held the Computational Intelligence Application Workshop, where reports were discussed in more detail in the field of computational intelligence, modern computer technologies and the development of smart systems, as well as practical developments from this research topic were demonstrated. Current issues of information technologies and smart systems were discussed, analysis of modern research and prospects for the development of these directions was carried out. This contributed to the expansion of cooperation between invited guests, speakers and listeners of the event.

It is planned that the Conference will become annual in the future.

Information about the Conference is available online at https://science.lpnu.ua/ciss-2024

CONTENTS

**Conference materials are presented in alphabetical order according to the last name of the first author*

**Conference materials are presented in alphabetical order according to the last name of the first author*

Maksym Arzubov[1], Mariia Markiv[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, marzubov@gmail.com
[2]Lviv Polytechnic National University, Lviv, Ukraine, mariia.markivv@gmail.com

CAR RENTAL MANAGEMENT SYSTEM OPTIMIZATION USING GEOSPATIAL DATA

Introduction. In the rapidly evolving landscape of urban development, managing transportation efficiently, particularly car rentals, is crucial for promoting urban mobility and reducing environmental impacts. The adoption of geospatial analysis in this sector has opened new avenues for enhancing the efficiency of car rental operations. This is especially relevant in the context of urbanization, where it is important to ensure efficient car placement planning, reducing operational costs and increasing customer satisfaction [1]. Geospatial analysis allows companies to obtain accurate data on the places of start and end of trips, which contributes to the development of strategies for the optimal location of resources.

This thesis explores the integration of geospatial analysis with clustering techniques and predictive analytics to develop a car rental management system tailored for modern urban environments [3]. The goal is to improve operational efficiency, enhance customer satisfaction, and decrease environmental footprints through strategic vehicle placement and resource management.

The objectives of this research are to design, develop, and implement a sophisticated system that employs geospatial analysis, data clustering, and predictive forecasting to optimize the deployment and management of car rentals in urban settings [6]. The system aims to provide a scalable solution that can adapt to varying urban dynamics and customer demand patterns, ensuring that services are both efficient and cost-effective.

Materials and methods. To meet these objectives, data collection involved compiling extensive datasets from the company's historical records on car usage and real-time data feeds detailing current rental demand. This comprehensive dataset forms the backbone of our geospatial analysis, which was conducted to map the most frequented routes and identify key points of demand throughout the city [4]. This phase of geospatial analysis was crucial as it helped in understanding the spatial relationships and patterns that are critical in planning the strategic placement of rental vehicles [4]. The insights gained from this analysis provided a data-driven foundation for the subsequent application of clustering algorithms.

Data collection involved compiling extensive datasets from the company's historical records on car usage and real-time data feeds detailing current rental demand. Geospatial analysis was conducted to map the most frequented routes and identify key points of demand throughout the city. This analysis helped in understanding spatial relationships and patterns which are critical in planning the placement of rental vehicles.

Clustering algorithms were then applied to segment different areas of the city based on usage patterns, demand density, and customer preferences [2]. This segmentation allowed for tailored marketing strategies and customized service offerings, enhancing customer engagement and satisfaction. The forecasting models utilized historical and current data to predict future demand trends. These predictions were instrumental in making informed decisions regarding fleet management, vehicle deployment, and operational scheduling.

To assess the effectiveness of various libraries for clustering data and selecting the optimal solution for our car rental system, a series of experiments were conducted. The main libraries examined were Leaflet, Mapbox, and Google Maps. Each library has unique features, advantages, and drawbacks, which were thoroughly studied and compared.

Before conducting the experiments, key evaluation criteria were established, including:

- Clustering Time: Time required to perform clustering on the data.
- Clustering Accuracy: Measured using the Dunn Index where higher values indicate better clustering by maximizing inter-cluster distance and minimizing intra-cluster distance.
- Performance: Efficiency when handling large data volumes.

- Visualization Capabilities: Quality and flexibility of data visualization.
- User Interface: Ease of use for the end-users.
- Integration Ease: How easily the library integrates with other system components.
- Support and Documentation: Availability and quality of technical support and documentation.

A testing environment was set up for each library using the same dataset comprising the start and end coordinates of rental car trips. This allowed for objective comparison under consistent conditions.

Table 1. Experiment 1: Clustering with 1,000 records

Library	Clustering Time (sec)	Dunn Index	Performance	Visualization	Integration Ease	Notes
Leaflet	2.5	0.75	Moderate	Limited	Easy	
Google Maps	2.2	0.78	High	Excellent	Difficult	Licensing limitations
Mapbox	1.8	0.78	High	Customizable	Easy	Best overall performance

Table 2. Experiment 2: Clustering with 3,000 records

Library	Clustering Time (sec)	Dunn Index	Performance	Notes
Leaflet	7.5	0.7	Decreased	Performance drops with larger data size
Google Maps	6.6	0.72	High	Integration complexity remains
Mapbox	5.5	0.8	High	Consistently superior performance

Table 3. Experiment 3: Clustering with 5,000 records

Library	Clustering Time (sec)	Dunn Index	Notes
Leaflet	12.5	0.68	Significant decrease in performance
Google Maps	10.8	0.7	High performance but integration issues
Mapbox	9.2	0.78	Maintains high performance

The experiments conducted provide a comprehensive understanding of each library's capabilities and suitability for the project. Mapbox emerged as the optimal choice due to its high performance, accuracy in clustering, superior visualization capabilities, ease of integration, and robust support and documentation. Its performance was consistent across different data sizes, and it provided the best user experience due to its customizable visualization features and straightforward integration with other system components. Thus, Mapbox is recommended for implementing the car rental management system's clustering functionality to ensure efficient data processing and user-friendly interfacing.

Results. The practical value of this work is to provide tools for the dynamic management of a network of rental points, grouping them into clusters that respond to real geographic and demographic characteristics of demand. This allows businesses not only to increase profitability but also to be more flexible and adaptable in today's fast-changing business environment. System advanced analytical capabilities allow you to monitor trends and adapt the vehicle placement strategy to current and future market needs, thereby ensuring the company's sustainable growth and high level of customer satisfaction.

In addition to the basic functions, the developed system also includes an advanced clustering capability that uses not only current but also historical data from the operational records of the car rental company. Thanks to this function, the system can analyze trends and changes in usage patterns and observe the evolution and movement of clusters of users over time. This allows not only to respond to the current needs of the market but also to adapt the strategic plan for the development of the car rental network, ensuring an increase in income and meeting customer requirements. This possibility of periodically updating data and forecasting future trends opens up ways to optimize the allocation of resources and improve customer service.

The implementation of this system demonstrated significant improvements in several operational metrics. Strategic car placement, guided by insights from geospatial analysis, led to a 30% reduction in customer wait times. Improved vehicle allocation strategies, informed by clustering and demand forecasts, reduced fleet operational costs by 20%. Additionally, the system's ability to meet customer expectations more accurately resulted in a 25% increase in customer satisfaction scores.

Discussion of the findings highlights the system's capacity to transform car rental services by providing a data-driven approach to fleet management [5]. The integration of geospatial analysis, clustering, and forecasting not only enhances operational efficiencies but also supports sustainable practices by reducing unnecessary vehicle idle times and optimizing the use of resources.

Conclusions. This research underscores the potential of geospatial technologies to revolutionize car rental services. It provides a framework for other urban service providers to incorporate similar technologies for better resource management. Looking forward, the research could explore the incorporation of real-time traffic data and advanced machine learning algorithms to further enhance the accuracy of demand forecasting and dynamic pricing models. Expanding the system to integrate electric vehicle data could also support broader sustainability goals by facilitating the transition to greener urban transport solutions.

References in this thesis draw on contemporary studies by John Doe and Anna Smith on the implications of geospatial analysis in urban transport management, as well as Richard Roe's research on the use of clustering techniques for dynamic fleet management. These studies provide a theoretical foundation for the applied methodologies and validate the relevance of the research in current urban development contexts.

1. Brito, B., Costa, D. G., & Silva, I. (2024, June). Geospatial Risk Assessment of Cyclist Accidents in Urban Areas: A K-Means Clustering Approach. In *2024 IEEE 22nd Mediterranean Electrotechnical Conference (MELECON)* (pp. 744-749). IEEE. https://doi.org/10.1109/MELECON56669.2024.10608791
2. Geetha, K., Srivani, A., Gunasekaran, S., Ananthi, S., & Sangeetha, S. (2023, September). Geospatial Data Exploration Using Machine Learning. In *2023 4th International Conference on Smart Electronics and Communication (ICOSEC)* (pp. 1485-1489). IEEE. https://doi.org/10.1109/ICOSEC58147.2023.10275920
3. Kiani, B., Fatima, M., Amin, N. H., & Hesami, A. (2023). Comparing geospatial clustering methods to study spatial patterns of lung cancer rates in urban areas: A case study in Mashhad, Iran. *GeoJournal*, *88*(2), 1659-1669. https://doi.org/10.1007/s10708-022-10707-3
4. Thakur, A. (2021). Car rental system. *International Journal of Research Applications in Science, Engineering and Technology*, *9*(7), 402-412. https://doi.org/10.22214/ijraset.2021.36339
5. Turoń, K. (2023). Car-sharing systems in smart cities: a review of the most important issues related to the functioning of the systems in light of the scientific research. *Smart Cities*, *6*(2), 796-808. https://doi.org/10.3390/smartcities6020038
6. Yang, Y., Jin, W., & Hao, X. (2008, October). Car rental logistics problem: A review of literature. In *2008 IEEE International Conference on Service Operations and Logistics, and Informatics* (Vol. 2, pp. 2815-2819). IEEE. https://doi.org/10.1109/SOLI.2008.4683014

Maksym Arzubov[1], Ivanna Ksondzyk[2]
[1] Lviv Polytechnic National University, Lviv, Ukraine, m.arzubov@gmail.com
[2] Lviv Polytechnic National University, Lviv, Ukraine, imksoiva@gmail.com

SMART CITY SOUND LANDSCAPE SYNTHESIS SYSTEM

Introduction. In the era of modern technologies and the rapid development of information systems, the significance of integrating technical means into the life of society cannot be overstated. In particular, smart cities are one of the key trends in the development of the modern world [4]. The concept of a "smart city" is becoming increasingly relevant in the context of solving a wide range of problems, from optimizing transportation infrastructure to ensuring the safety and comfort of residents. Therefore, addressing noise pollution is a pressing task for modern cities [3].

This research aims to design and implement a system that effectively collects, processes, and analyzes acoustic data within urban environments using advanced information technologies. The focus of the project is on the processes involved in developing a sound landscape synthesis system for smart cities. This system classifies acoustic data gathered from various city locations, providing detailed insights into the detected soundscape.

The research explores various methods and algorithms for processing this data and constructing sound classification models, alongside techniques for visualizing noise levels and identifying their sources. This comprehensive approach ensures a robust framework for managing urban acoustic environments.

Materials and methods. The project collected diverse urban sounds from open sources processed through techniques such as noise filtering and amplitude normalization to ensure data quality. A set of models was implemented: traditional classifiers [2] (e.g., XGBClassifier) for basic sound pattern recognition and a CNN [1, 5] that uses spectrograms for a more detailed analysis. The models were selected and validated based on their performance metrics, such as accuracy and f1-score.

The UrbanSounds dataset, containing over 8,000 urban sounds, was selected for model training. The data are classified into 9 different categories and are practically balanced.

To choose the model to be used in the system, the training results of several classifiers were analyzed, employing methods such as decision trees, random forests, adaptive boosting, extreme boosting, gradient boosting, k-nearest neighbors, and a neural network with hidden layers (Fig. 1).

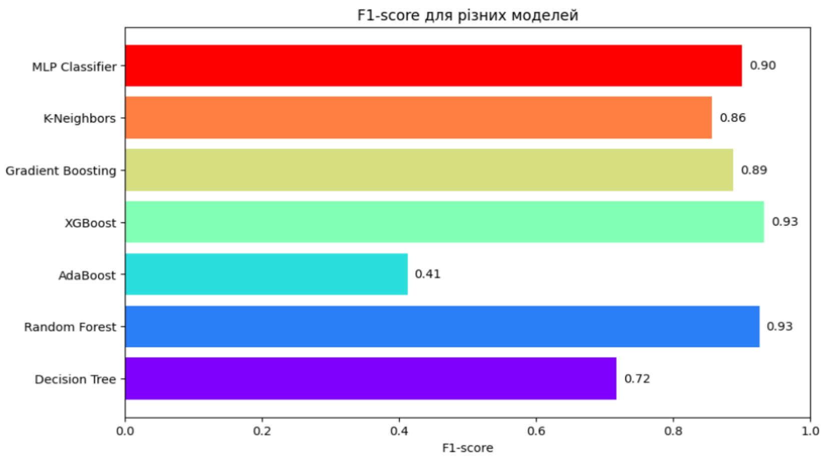


Fig. 1. Training results of classifiers

Next, validation was conducted, and based on the results, the XGBClassifier was selected for use in the system. After all training stages, the model achieved an f1-score of 94.44, indicating high accuracy in recognizing sound sources.

Since CNNs accept input data as images, it is necessary to convert the audio signal into this form. Mel-spectrograms, which visually represent the energy of the signal over time and frequency on a mel-scale, are used for this purpose.

Results. The neural network built consists of six convolutional layers (Conv2D), normalization layers (Batch Normalization), and dropout layers. The neural network was trained over 40 epochs with a batch size of 64, meaning the model updates its parameters after processing every 64 examples from the training set. The result of this training is a model accuracy of 95.3% and an f1-score of 95, which are higher metric values than those of traditional models.

The Streamlit platform was chosen to create the system interface, providing capabilities for developing interactive web applications, while being simple and fast to use. The visual component of the system includes three main blocks that define all user functionality.

The main part of the interface is an interactive map displaying all recorded sounds as bars in the locations of the sensors across the city (Fig. 2). Users can also filter the sounds they wish to display by date, time of day, and classes.

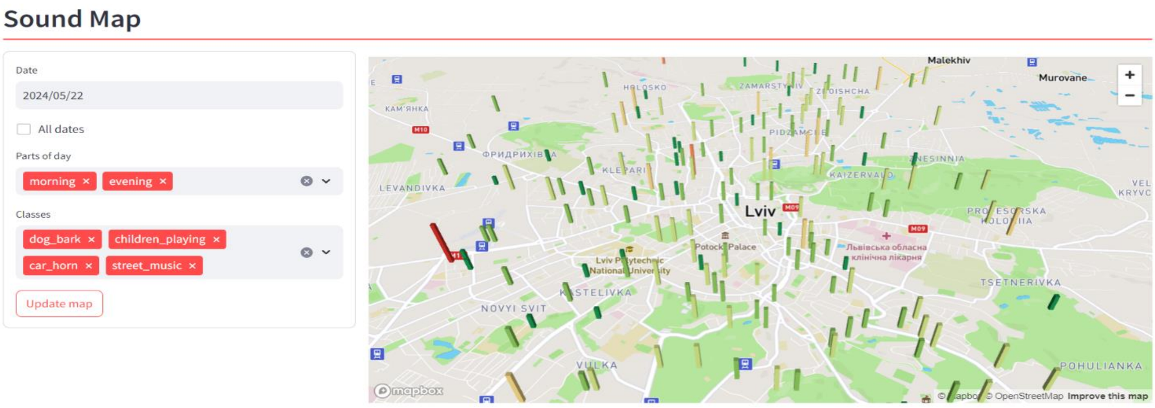


Fig. 2. Sound map display

Conclusions. The system serves as a crucial tool for local authorities to make informed decisions regarding urban planning and noise control. It also aids residents in selecting quieter neighborhoods, thereby improving their living conditions. Additionally, the system can be integrated with other smart city technologies, such as traffic management systems, air quality monitoring, and environmental monitoring systems, to create a comprehensive urban management solution that ensures the maximum well-being of city residents. The system flexibility allows for seamless integration with other smart city applications, such as traffic and environmental monitoring systems, creating a holistic urban management tool.

This research contributes to the smart city initiative by providing a robust framework for noise pollution monitoring and management. Future work could explore the integration of more diverse data sources and advanced predictive models to further enhance the system's accuracy and reliability.

1. Gourisaria, M. K., Agrawal, R., Sahni, M., & Singh, P. K. (2024). Comparative analysis of audio classification with MFCC and STFT features using machine learning techniques. *Discover Internet of Things*, 4(1). https://doi.org/10.1007/s43926-023-00049-y.
2. Hsieh, W. W. (2023). Decision trees, random forests and boosting. In Cambridge University Press eBooks, 473–493. https://doi.org/10.1017/9781107588493.015
3. Ranpise, R. & Tandel, B. (2022). Urban road traffic noise monitoring, mapping, modelling, and mitigation: A thematic review. *Noise Mapping*, 9(1), 48-66. https://doi.org/10.1515/noise-2022-0004
4. Son, T. H., Weedon, Z., Yigitcanlar, T., Sanchez, T., Corchado, J. M., & Mehmood, R. (2023). Algorithmic urban planning for smart and sustainable development: Systematic review of the literature. *Sustainable Cities and Society*, 94, 104562. https://doi.org/10.1016/j.scs.2023.104562
5. Yin, T. (2023). Music track recommendation using Deep-CNN and MEL Spectrograms. Mobile Networks and Applications. https://doi.org/10.1007/s11036-023-02170-2

Viktoriia Badz[1], Vasyl Teslyuk[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, viktoriia.m.badz@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, vasyl.m.teslyuk@lpnu.ua

APPROACH FOR DETERMINING AUTHORSHIP OF ENGLISH-LANGUAGE TEXTS USING CONVOLUTIONAL NEURAL NETWORKS

Introduction. In the digital age, where vast amounts of textual content are generated and shared online, the need for reliable methods of determining authorship has become increasingly important. Identifying the creator of a text is not only crucial for preserving intellectual property but also plays a significant role in areas such as cybersecurity, academic integrity, and digital publishing. Traditional approaches to authorship attribution often rely on manual feature extraction, which can be labor-intensive and limited in accuracy when dealing with large and complex datasets. With the advancements in artificial intelligence, Convolutional Neural Networks (CNNs) — a type of deep learning model — have gained prominence for their exceptional performance in various pattern recognition tasks, particularly in image classification. This success has led researchers to explore the application of CNNs in the field of natural language processing for authorship determination. By leveraging CNNs' ability to automatically detect intricate patterns in data, this approach holds the potential to revolutionize the way authorship is attributed, offering greater accuracy and scalability compared to traditional methods. This research explores the development and implementation of a CNN-based model for determining the authorship of English-language texts. It highlights the methodological advancements brought by CNNs in capturing the stylistic nuances of different authors and demonstrates how these models outperform conventional techniques in tasks requiring the analysis of large corpora and multi-class classification. Additionally, the paper discusses the practical applications of this technology in various fields, including academic plagiarism detection, cybersecurity, and literary studies, underscoring its growing relevance in today's digital landscape.

The relevance of the research. In the world of modern technologies, where information is stored online and various means of digitizing paper resources are increasingly used, the problem of losing data about the author of the created text becomes important. Therefore, modern approaches of using artificial neural networks (which demonstrate effectiveness in various fields of application) to recognize the author's styles and, on its basis, determine the authorship of the text are increasingly finding their application and development in this subject area. The use of Convolutional Neural Networks (CNNs) for the tasks of image recognition and classification shows high efficiency compared to other approaches [1 – 4]. Therefore, the idea arose to use Convolutional Neural Networks for the problem of determining the authorship of texts. The main components of this approach are: the stage of selection of individual characteristics that affect the author's writing style and, in fact, the training of the model.

The object of research is the process of determining the authorship of English-language texts using CNNs. The subject of the study is models of automation of the author attribution process using convolutional neural networks. The relevance of CNN-based authorship attribution research extends far beyond linguistics and literature into fields like cybersecurity, law enforcement, academic integrity, and digital publishing. This approach allows for more sophisticated, accurate, and scalable analysis of textual data, making it a valuable tool in a wide variety of modern applications. With the growing importance of digital content, the need for reliable authorship attribution methods is only becoming more urgent, and CNNs are at the forefront of this evolving landscape. The scientific novelty of this CNN-based approach lies in its use of a powerful deep learning architecture to automatically detect intricate writing patterns, its adaptability to large datasets and complex texts, and its potential to extend authorship attribution into different areas like cross-linguistic plagiarism detection, cybersecurity, and law enforcement. This represents a significant advancement over traditional methods that rely on manual feature extraction and limited datasets.

The methods and materials. The methods and materials used for determining authorship of English-language texts based on CNNs involve several key stages, including data preparation, model architecture, training, and evaluation. The model for determining text authorship based on Convolutional Neural Networks is a modern approach in the field of natural language processing (NLP). This method uses the powerful capabilities of CNNs to capture and analyze local patterns in textual data, which can be useful for identifying the author's style. The main steps and components of such a model include: data preparation, text vectorization, Convolutional Neural Network architecture, model training and evaluation, model tuning and optimization. At the stage of data preparation, it is necessary to collect a corpus of texts by different authors (these can be articles, books, blogs or other types of texts); after that, the texts are analyzed and cleaned: removal of extra characters, removal of stop words, tokenization of the text, reduction of all text to lower case; after that, it is necessary to divide into training and test samples to check the quality of the model. The second stage: using vectorization techniques to convert words into vectors; construction of vector sequences — conversion of text into sequences of vectors representing each word. The third stage: designed to develop the CNN architecture. The input layer accepts a sequence of vectors. Convolutional layers use filters to detect local patterns in text; multiple convolutional layers can be applied for deeper analysis. Dense layers are responsible for combining features highlighted by convolutional layers and making a final decision. Output layer: usually a softmax for multi-class classification: identifying an author from a set of possible authors. Python is used for implementing the CNN model due to its extensive libraries for machine learning and natural language processing. TensorFlow provides a high-level API for building and training CNN models. NLTK is useful for preprocessing and tokenization of text. Scikit-learn is used for model evaluation metrics like precision, recall, F1-score, and confusion matrix.

The results of research and their practical value. The results of research on using a CNN-based approach for determining authorship of English-language texts have demonstrated both promising accuracy and practical value. CNNs effectively capture stylistic nuances through convolutional filters, which excel at identifying n-grams (sequences of words) that are unique to specific authors. This capability helps improve attribution in longer texts like articles, essays, and novels. The created CNN-based model has performed well when attributing authorship in cases where there are numerous possible authors (multi-class classification), producing higher accuracy than traditional methods in these scenarios. Unlike traditional methods that require manual feature extraction (e.g., average sentence length, vocabulary richness), CNNs automatically learn and extract relevant features from raw text. This leads to more generalized and scalable models.

The CNN-based approach for authorship determination requires a combination of well-prepared datasets, deep learning architectures, and rigorous evaluation methods. By leveraging the pattern-detection capabilities of CNNs, this method allows for nuanced and accurate authorship attribution, especially in large and complex corpora.

The CNN-based model produced a significant improvement in accuracy compared to traditional methods of authorship attribution. The study found that CNNs, due to their ability to capture local patterns in textual data, are especially effective in multi-class classification scenarios where numerous possible authors are involved.

The CNN model achieved an accuracy rate higher than 90% in correctly identifying authors from a set of possible candidates. This surpasses the accuracy typically achieved by classical methods, such as Naive Bayes or Support Vector Machines (SVMs), which often fall in the range of 70–85% depending on the dataset size and complexity. The convolutional layers in CNNs excel at detecting n-grams (sequences of words) that are unique to an author’s writing style. This ability helped the model to effectively differentiate between writing styles based on subtle patterns in word usage, sentence structure, and syntax. The study analyzed how the CNN filters detected n-grams within the range of 3-5 words, which are typically most indicative of stylistic patterns. The model was able to automatically learn these patterns without manual feature engineering, increasing its adaptability and efficiency in different text genres. CNNs proved particularly effective in longer texts, such as essays, articles, and novels. This contrasts with traditional approaches, which tend to lose accuracy with increasing text length due to limitations in feature extraction. In the study, the model was tested on texts ranging from 500 to 5,000 words. The CNN-based model maintained high performance across these lengths.

The created CNN-based model can be deployed to detect plagiarism by verifying whether a piece of text matches the writing style of the claimed author. This is particularly valuable in academic and educational institutions, where plagiarism detection tools can be improved with CNNs' ability to discern subtle writing patterns. CNNs have the potential to handle translated works, helping to identify plagiarism across languages by recognizing an author's stylistic patterns regardless of translation. In cybersecurity and law enforcement, CNNs can be used to attribute authorship to anonymous or pseudonymous texts, such as online messages, posts, or threatening letters. This aids in criminal investigations and digital forensics by linking anonymous communications to known individuals based on their writing style. CNN models have practical value in literary studies for resolving disputes over the authorship of historical texts. CNNs can assist in identifying the true origin of online news articles and distinguishing between legitimate reporting and fake news by analyzing the writing style of the author.

Conclusions. The model has been developed for determining the authorship of English-language texts using Convolutional Neural Networks, the main elements of the model and their characteristic features are presented. The analysis of the proposed approach was carried out, and the main advantages and disadvantages were determined. The model for determining text authorship based on Convolutional Neural Networks is a powerful tool in the modern field of NLP, capable of detecting subtle stylistic features of authors in texts. With proper data preparation and model tuning, CNNs can significantly outperform traditional text analysis methods in authorship attribution tasks. CNNs offer a powerful tool for authorship attribution due to their ability to extract and recognize complex patterns in texts. However, they require a well-structured and sufficiently large dataset for training, and their performance is sensitive to text variability and genre.

1. Hussain, M., Bird, J. J., & Faria, D. R. (2018). A Study on CNN Transfer Learning for Image Classification. *Advances in Computational Intelligence Systems*, 191–202. https://link.springer.com/chapter/10.1007/978-3-319-97982-3_16
2. Dzakirizvi, M. (2024). Image Classification Using CNN. *Analytics Vidhya*. https://www.analyticsvidhya.com/blog/2020/02/learn-image-classification-cnn-convolutional-neural-networks-3-datasets/
3. Purwono, P., Ma'arif, A., Rahmaniar, W., Fathurrahman, H. I. K., Frisky, A. Z. K., & Haq, Q. M. (2023). Understanding of Convolutional Neural Network (CNN): A Review. https://doi.org/10.31763/ijrcs.v2i4.888

Convolutional Neural Networks for Authorship Attribution of Short Texts (2017). *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*: Volume 2, Short Papers, pp 669–674. https://doi.org/10.18653/v1/E17-2106

Volodymyr Bak[1], Anastasiia Doroshenko[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, volodymyr.bak.kn.2021@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, anaastasiia.v.doroshenko@lpnu.ua

CHALLENGES AND PROBLEMS OF PERSONAL DATA SECURITY IN AR AND VR SYSTEMS

Introduction. Augmented reality (AR) and virtual reality (VR) systems are reshaping how individuals interact with digital content by merging physical and virtual worlds. These technologies are widely used in industries such as gaming, education, healthcare, and business, facilitating real-time interactions and offering immersive experiences [1]. However, with the rise of these systems comes an urgent need to address the protection of personal data. AR and VR applications often collect vast amounts of user data, including behavioral patterns, location, and biometric data, which raises concerns about privacy and the potential misuse of sensitive information.

Problem Statement. Privacy issues in AR and VR systems have become a primary concern for users. Research in [2] user perceptions revealed that between 7% and 51% of participants suggested "limiting data collection" as a strategy to protect their privacy. Methods included turning off device sensors, using devices only in low-privacy areas, or disconnecting them when not in use. Additionally, some users expressed a desire for greater control over their data, emphasizing that data should only be anonymized and used for user benefit, without being repurposed for targeted advertising or manipulation.

A significant portion of users is also concerned about the emotional data collected by AR and VR systems. Emotional data, such as mood and emotional state, can reveal sensitive information that, if compromised, could be used for social engineering or manipulation. Users worry about being constantly monitored and have expressed unease about how such data could affect their daily lives.

Main Part. Several approaches and solutions have emerged to address the privacy and security challenges in AR and VR. These include strategies to limit data collection, enhance user control over data, and improve security through authentication technologies.

Public Awareness and Control is one of the key principles of personal data security in AR/VR systems. One of the primary methods users employ to protect their privacy is reducing device usage or adjusting how they use the device. Around 9% of participants from the study [3] expressed the need for more effective control over data. They sought detailed knowledge about data collection and transparency in privacy policies. These actions suggest a demand for companies to improve how they inform and empower users regarding their data.

The collection of emotional data presents both risks and opportunities. On the one hand, emotional data could be exploited for manipulative purposes. As is well known, emotional data, according to the GDPR, are considered sensitive data that require particularly strong protection. On the other hand, it holds potential for improving diagnostics and treatment of psychological disorders. Future advancements could allow for more controlled and secure use of emotional data, where data is anonymized and shared only for medical or research purposes.

Security solutions represent the most crucial aspect in addressing the challenges of data protection in AR and VR systems. These solutions must ensure robust data protection mechanisms, including encryption, authentication, and privacy-preserving technologies, to safeguard sensitive user information [4]. Without effective security measures, AR and VR systems are vulnerable to data breaches and misuse, compromising user privacy and trust. To ensure the most effective protection of personal data in augmented and virtual reality systems, it is advisable to use a set of measures. In particular, the most widely used today are:

- Risk Assessment Frameworks. Research in [2] emphasized the use of risk frameworks such as attack trees to calculate risk scores and assess vulnerabilities in AR and VR systems. These frameworks can help identify and mitigate risks before they become significant security threats.
- Data Obfuscation Techniques. Methods such as digital watermarks and steganography have been proposed to hide sensitive information in AR/VR environments. These techniques help protect

the privacy of users by making it more difficult for unauthorized parties to access or manipulate data. However, some vulnerabilities still exist, especially in complex attacks, and further research is needed to improve these methods.

- Authentication Techniques:
  - Knowledge-Driven Biometric Authentication. RubikBiom, a biometric authentication method that uses user behavior, has demonstrated high accuracy (98.91%) in verifying identity. It leverages knowledge of user movement patterns during interaction, making it a reliable tool for enhancing security [5].
  - RubikAuth. Another experimental authentication system, RubikAuth, uses a unique method of entering PIN codes in a 3D environment. The system has shown high success in preventing security breaches, with 98.52% of attacks being unsuccessful [6].
  - BioMove. This technique uses users' kinesiological movements for biometric authentication. It has been successful in 95% of tests, though similar physical characteristics between users can occasionally lead to compromised security [7].

Combining various authentication technologies, such as RubikAuth and BioMove, could create a more robust and efficient system. The integration of these systems would reduce security vulnerabilities while maintaining quick authentication times, ensuring that AR and VR systems are both secure and user-friendly. A schematic example is presented on Figure 1.

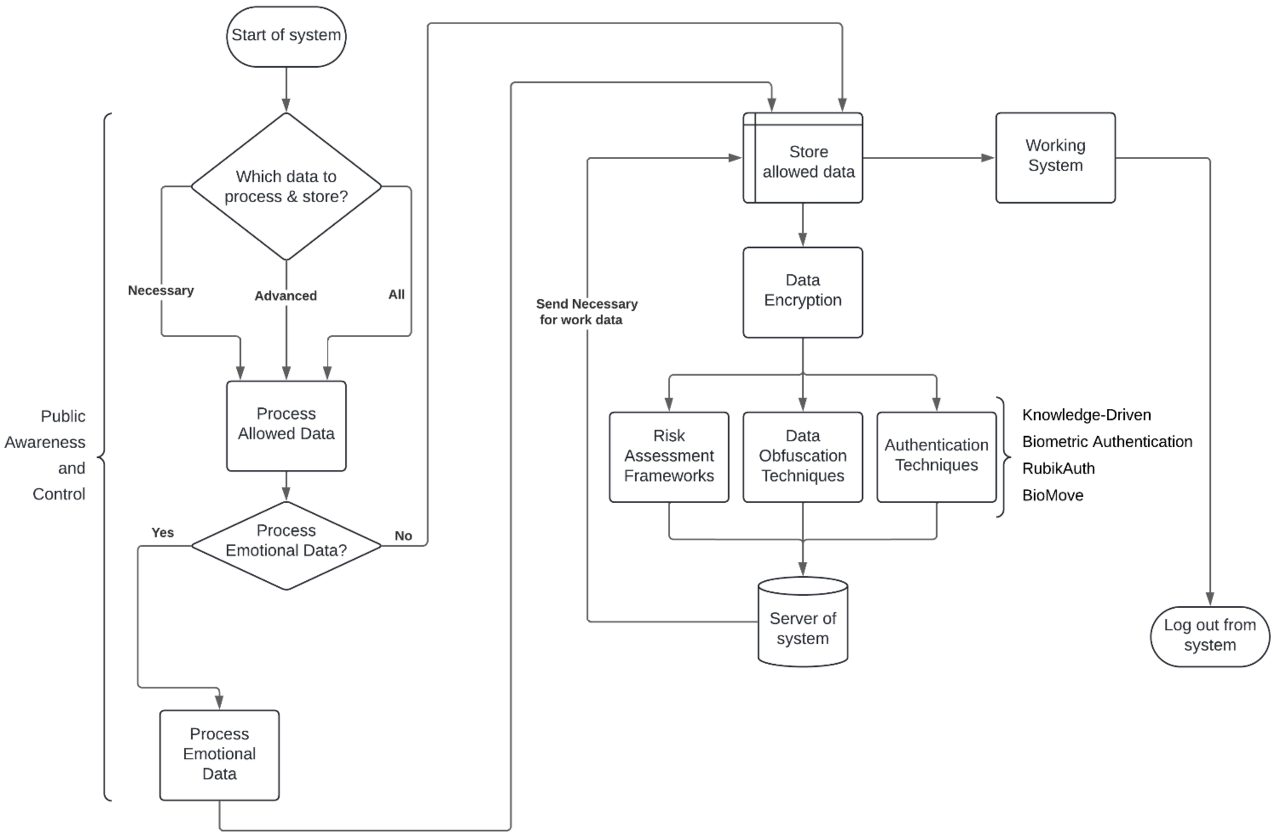


Fig. 1. Process diagram for AR/VR system

Conclusion. As AR and VR technologies continue to evolve, addressing privacy and security concerns becomes more pressing. Current solutions, such as risk assessment frameworks, data obfuscation techniques, and advanced biometric authentication, have shown promise in mitigating these issues. However, as these technologies become more pervasive, further research and development are needed to ensure that personal data, particularly emotional data, remains secure. Future advancements should focus on enhancing user control, improving transparency, and developing even more sophisticated security measures to protect users in immersive environments.

Acknowledgment. This work was realized within the framework of the Erasmus+ Jean Monnet Chair 2022 «101085612 – DataProEU - Data Protection in the EU».

1. Teslyuk, V. M., Gado, I. V., Havryliak, T. V., & Paziuk, O. V. (2024). Model and tools of the navigation system between educational institution buildings using augmented reality technology. Ukrainian Journal of Information Technology, 6(1), 86–94. https://doi.org/10.23939/ujit2024.01.086
2. Valluripally, S., Gulhane, A., Mitra, R., Hoque, K. A., & Calyam, P. (2020). Attack Trees for Security and Privacy in Social Virtual Reality Learning Environments. *2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*, 1–9. https://doi.org/10.1109/CCNC46108.2020.9045724
3. Hadan, H., Wang, D. M., Nacke, L. E., & Zhang-Kennedy, L. (2024). Privacy in Immersive Extended Reality: Exploring User Perceptions, Concerns, and Coping Strategies. *Proceedings of the CHI Conference on Human Factors in Computing Systems*, 1–24. https://doi.org/10.1145/3613904.3642104
4. Hlod, S., & Doroshenko, A. (2021). Application of augmented reality technologies for education. *2021 IEEE 16th International Conference on Computer Sciences and Information Technologies (CSIT)*, 159–162. https://doi.org/10.1109/CSIT52700.2021.9648783
5. Mathis, F., Fawaz, H. I., & Khamis, M. (2020). Knowledge-driven Biometric Authentication in Virtual Reality. *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, 1–10. https://doi.org/10.1145/3334480.3382799
6. Mathis, F., Williamson, J. H., Vaniea, K., & Khamis, M. (2021). Fast and Secure Authentication in Virtual Reality Using Coordinated 3D Manipulation and Pointing. *ACM Transactions on Computer-Human Interaction*, *28*(1), 1–44. https://doi.org/10.1145/3428121
7. Olade, I., Fleming, C., & Liang, H.-N. (2020). BioMove: Biometric User Identification from Human Kinesiological Movements for Virtual Reality Systems. *Sensors*, *20*(10), 2944. https://doi.org/10.3390/s20102944

Oleksandr Basalkevych[1], Denys Rudavskyy[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, oleksandr.a.basalkevych@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, denys.v.rudavskyi@lpnu.ua

FLEXIBLE SOFTWARE ARCHITECTURE FOR WIND TURBINE BLADE DAMAGE CALCULATION

Introduction. Wind energy has become one of the most relevant directions in the field of renewable energy sources. With the development of technologies and increasing awareness of the need for environmentally clean energy sources, it has turned into a key element of the global energy strategy. Each year, the total capacity of wind power stations consistently increases, indicating a growing interest in this type of energy among both developed countries and private investors.

Approximately 20% of wind turbine accidents are attributed to wind turbine blade failures [1], with crack-like defects, particularly fatigue cracks, considered the most dangerous, as they act as stress concentrators and can rapidly propagate during the product's operation [2]. Longitudinal and transverse cracks in the root sections of wind turbine blades are primarily caused by high stresses induced by geometric changes and fatigue loading in this area [3].

The purpose of this publication is to propose a flexible scheme with the possibility of its further improvement by the researcher for the automated numerical calculation of damage nucleation in the blade root using the rain flow method [4].

Materials and methods. The main inputs for the system consist of a sample of historical data on wind speed, rotor rotation speed, and blade pitch angle relative to its axis obtained from appropriate sensors. Secondary input parameters can include ambient temperature, air humidity, etc.

During the initial data preprocessing stage, it is necessary to perform outlier filtering – identifying values that significantly deviate from the mean. These can be caused by voltage spikes in sensors, electronic noise and interference, abrupt changes in the ambient environment, and so on.

In the second stage, it is necessary to determine the functional relationship between input parameters and the load on the wind turbine blade root:

$$L_i = f(M_i, A_i),$$

where M_i and A_i are the values of primary and secondary input parameters at time i, L_i is the load value at the turbine blade root at time i.

In the third stage, the load values become input data for the Rainflow-counting method, which transforms the obtained variable chaotic signals into a set of cycles with consistent amplitude and frequency. This method makes it possible to single out fatigue regular cycles for their further analysis in order to assess the fatigue damage of materials and components. Thus, the initial data of the Rainflow-counting method is a set of pairs of values (L_k^*, N_k), where L_k^* is the value of the amplitude of the k-th load, N_k is the number of regular cycles of the k-th load.

The final stage of the calculation involves using the Miner's rule [5] for linear accumulation of damage. Its main concept lies in the direct proportionality of damage accumulation in the material to the applied load and the number of load cycles. By summing up the damage caused by each cycle, one can evaluate the total defect accumulated in the material and predict potential failure points. Thus, the input data for the fourth stage of the experiment is a set of pairs (L_k^*, N_k), obtained in the previous stage. Additionally, for the application of Miner's rule, the presence of an S-N curve describing the fatigue properties of the specific polymer material used in the wind turbine blade is necessary. The decision regarding the potential existence of a fatigue crack in the operational volume of the material is made based on calculating the parameter R according to the Miner's rule:

$$R = \sum_{i=1}^{k} \frac{N_i}{K_i},$$

where N_i is the number of load cycles at the load level L_i^*, and K_i is the maximum permissible number of cycles for the load level L_i^*, determined using the S-N curve.

The assumption of the presence of a fatigue crack in the root of the wind turbine blade can be made if the obtained value of R is greater than or equal to one (Fig. 1).

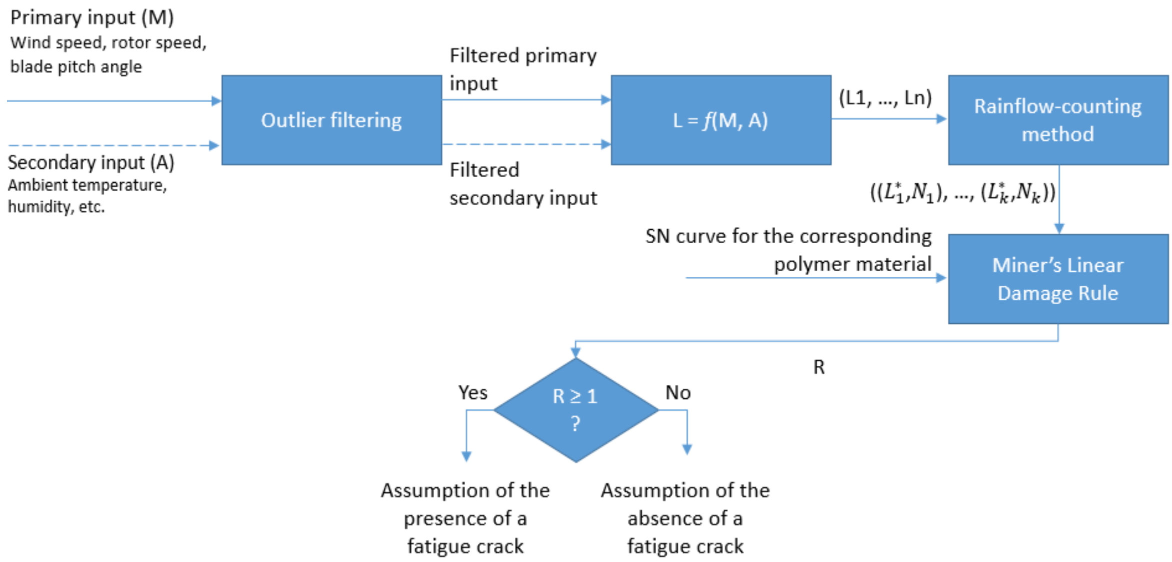


Fig. 1. The general decision-making scheme for determining the initiation of a fatigue crack in the root of a wind turbine blade

Results. The software implementation of such a simple logic is usually carried out using a straightforward Transaction Script pattern [6]. It involves the implementation of a sequence of actions in a procedural style. The main advantage is simplicity, while its disadvantage is the complexity of maintenance when expanding the system and rapidly adding new business logic. Fig. 2 presents the flowchart for the proposed numerical experiment.

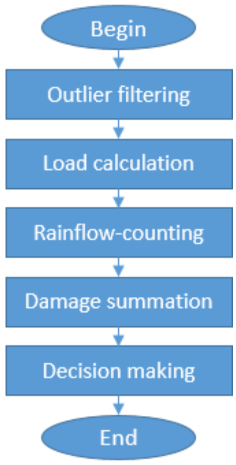


Fig. 2. Flowchart of the transaction script for the numerical experiment to determine the initiation of fatigue cracks at the root of a wind turbine blade

The scheme of the numerical experiment presented in this article represents a set of well-known methods for solving such fatigue accumulation problems. In the course of further research, it involves continuous improvement by adding and comparing alternative models of damage accumulation, filtering methods, and ways to determine the load dependence on historical input data to minimize error and consider the maximum number of parameters influencing the system. In other words, a rapid increase in complexity and enhancement of the basic implementation is anticipated, using alternative approaches for most stages of the experiment. Therefore, the use of the Transaction Script pattern is deemed impractical.

An alternative to the aforementioned approach is the use of a Domain Model architecture [6], which involves representing internal entities as objects that interact with and exchange data with one another. Its main advantage is flexibility and extensibility: the architecture makes it relatively easy to add new business logic and change behavior with minimal modifications to existing code. The modeling is more natural. An important drawback of this approach is the excessive complexity at the beginning of development. However, compared to the mentioned advantages, it is acceptable, considering the perspective of long-term system development.

Based on the Domain Model, a software solution architecture framework has been developed, providing flexibility in the choice of filtering methods, load determination, and decision-making regarding the potential presence of defects (see Fig. 3). This allows, based on the inherent potential for using polymorphic behavior of objects within the concept of object-oriented programming, to modify and present an enhanced implementation of the noise filtering algorithm and the function for load dependency on input historical data without altering the system architecture. If the Controller interacted with the system's objects directly, rather than through appropriate interfaces, replacing the aforementioned algorithms and methods would require significant modifications to the already written code, which contradicts the OCP (Open/Closed principle) of SOLID [6] and clearly indicates weak flexibility in the developed architecture. There is also an opportunity to improve the damage summation algorithm to the point of abandoning the use of the linear model since it is sufficiently simplified.

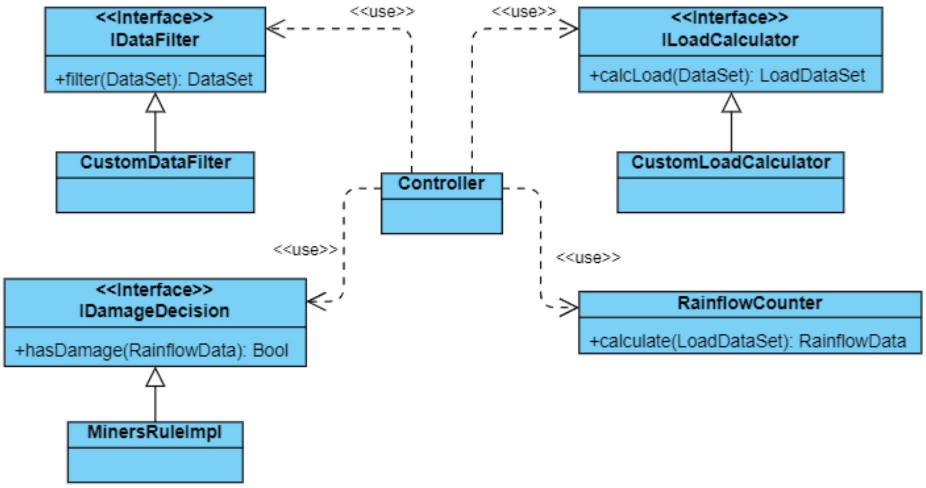


Fig. 3. UML class diagram of the general architecture of the software solution

Conclusions. Features of a flexible scheme for automated numerical calculation of fatigue damage accumulation in the blade root have been proposed and identified, with the potential for further improvement.

1. Wang, W., Xue, Y., He, C., Zhao, Y. (2022). Review of the Typical Damage and Damage-Detection Methods of Large Wind Turbine Blades. *Energies*, *15*(15), 5672. https://doi.org/10.3390/en15155672
2. Rudavskyi, D. V. (2011). *Residual resource of metal structural elements in Hydrogen-containing environments* [in Ukrainian]. Kyiv: Naukova Dumka.
3. Ataya, S.; Ahmed, M.M. (2013). Damages of wind turbine blade trailing edge: Forms, location, and root causes. *Engineering Failure Analysis*, *35*, 480–488. https://doi.org/10.1016/j.engfailanal.2013.05.011
4. Lee, Y., Barkley. M., Kang H.-T. (2012). Chapter 3 - Rainflow Cycle Counting Techniques. In *Metal Fatigue Analysis Handbook: Practical problem-solving techniques for computer-aided engineering* (pp. 89-114). Elsevier. https://doi.org/10.1016/B978-0-12-385204-5.00003-3
5. Lee, Y., Barkley. M., Kang H.-T. (2012). Chapter 9 - Vibration Fatigue Testing and Analysis. In *Metal Fatigue Analysis Handbook: Practical problem-solving techniques for computer-aided engineering* (pp. 333-382). Elsevier. https://doi.org/10.1016/B978-0-12-385204-5.00003-3
6. Martin, Robert C. (2003). *Agile software development: principles, patterns, and practices.* Pearson education Inc.

Oleh Berezsky[1], Pavlo Liashchynskyi[2], Petro Liashchynskyi[2]
[1] *West Ukrainian National University, Ternopil, Ukraine, olber62@gmail.com*
[2] *West Ukrainian National University, Ternopil, Ukraine, pavloksmfcit@gmail.com*

DIAGNOSTICS METHODS AND TOOLS IN MEDICAL SYSTEMS

Introduction. Over the past decades, computer-aided diagnostics has become a very promising field of research. Its main goal is to improve diagnostic and treatment procedures for radiologists and clinicians when analyzing medical images. With the help of big data and advanced artificial intelligence (AI) technologies, i.e. machine learning and deep learning algorithms, the healthcare system can be made more convenient, active, efficient, and personalized.

Artificial intelligence systems can diagnose diseases based on brief medical data and medical histories, even in cases with hidden specific symptoms. AI is already playing an important role in the early detection of diabetes [1] and breast cancer [2], as well as in many other cases with high accuracy.

Further innovations in this area will help to further improve the accuracy and efficiency of diagnostics, so research on this topic is an urgent task.

Computer-aided diagnosis systems (CAD) in the medical field can be considered as advanced expert and intelligent systems at the intersection of medicine and computer science. Such systems can use diagnostic rules to emulate the way a doctor makes a diagnosis. In this sense, such systems function as expert or decision support systems.

CAD is used to help doctors identify diseases by analyzing medical images. These systems can automatically detect signs of pathology and provide recommendations, which helps improve diagnostic accuracy and reduce the risk of human error.

Decision support systems (DSS) are computerized systems that help doctors and medical staff make informed decisions in the treatment and diagnosis process. They can analyze patient data, compare it with medical databases, make recommendations for diagnosis or treatment, and thereby increase the efficiency and accuracy of medical decisions. It should be understood that DSS are advisory in nature.

Analysis of CAD and DSS. The main purpose of CAD is to automate the process of detecting anomalies in medical images. DSS is broader in functionality and includes assistance in making decisions based on a large amount of data. This is the difference between the two systems.

In modern medicine, computer-aided diagnosis systems are becoming increasingly important due to their ability to improve diagnostic accuracy and increase treatment efficiency.

CAD systems include the following main components:

1. Data entry:
   - acquisition of medical images or patient data. This stage involves the collection of medical images using various medical devices such as X-ray machines, magnetic resonance imaging (MRI), computed tomography (CT) scanners, and ultrasound machines. Other patient data may also be used, such as blood test results or genetic data.
2. Image pre-processing:
   - filtering and correction of images to improve their quality. At this stage, the images are processed to remove noise, correct contrast, and other defects that may interfere with further analysis. This may include applying filters to reduce noise, equalize brightness and contrast, eliminate artifacts, and improve image clarity.
3. Segmentation:
   - highlighting characteristics or features that may be useful for diagnosis. This step involves analyzing images to identify specific features or characteristics, such as contours, textures, sizes, and shapes of objects. Feature extraction algorithms can use image segmentation techniques to highlight specific areas that may contain pathology.
4. Classification:
   - using algorithms to classify pathologies or normal conditions. At this stage, machine learning or artificial intelligence algorithms are used to analyze the selected features and classify them

as normal or pathological. Algorithms can be trained on large data sets to recognize common patterns of pathology, such as tumors, hemorrhages, or other abnormalities.

5. Visualization of the results:
   - providing results in a form convenient for the doctor. The results of the analysis are displayed in a form convenient for the doctor. This may include marking suspicious areas on the image, displaying statistical information about the probability of pathology, and providing recommendations for further action. It is important that the results are presented in such a way that the doctor can quickly and accurately interpret the information.

CAD architecture. These steps ensure the integral operation of the CAD system, which is aimed at improving the accuracy and efficiency of medical diagnostics.

We have developed our own CAD architecture (Fig. 1) that will be more flexible and easily adaptable to different types of medical images. It consists of three modules.

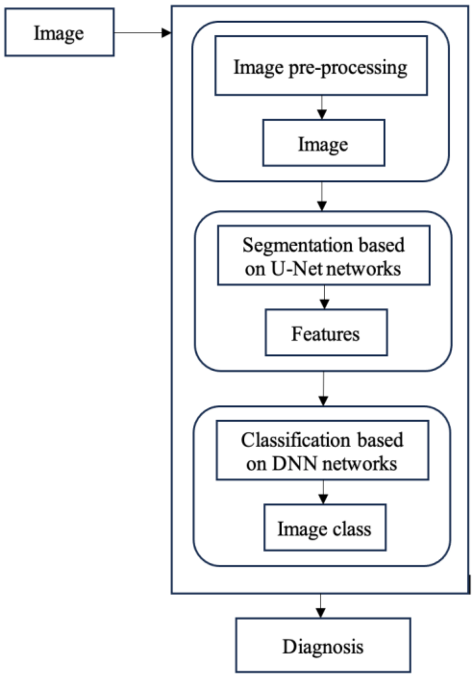


Fig. 1. Proposed CAD architecture.

As we can see in Figure, two of the three modules are based on neural networks, which are widely used for computer diagnostics.

Let's consider methods and tools for specific diagnostic tasks. They are shown in Table 1.

CNN architectures, such as Alex-Net [3], VGG-Net [4], and GoogleNet [5] have been developed and applied to various image recognition tasks. In general, CNN-based segmentation models can be divided into pixel-based and image-based approaches.

Table 1. Methods and tools for specific diagnostic tasks

Medical image analysis	Convolutional neural networks (CNN)	The most common for image analysis are VGGNet, ResNet, Inception
	Regional CNN	Used to detect objects in medical images
Analysis of electronic medical records	Recurrent neural networks (RNN)	Used to predict diseases based on patient history
	Graph neural networks (GNN)	They are used to analyze the relationships between various medical indicators
Analysis of genomic data	Deep neural networks (DNN)	Used to detect genetic mutations and predict susceptibility to certain diseases
	Machine learning based on trees	It is used for classification and prediction based on large genomic data sets

Image-based approaches, such as U-Net [6], take an image as input and produce a segmentation of the input image (the size remains the same). U-Net models have become popular due to their high efficiency and simplicity compared to pixel-based approaches.

Conclusion. Taking into account the above material, it was decided to use neural networks as the main tool for building a computer diagnostic system. This approach allows achieving high accuracy and reliability by effectively combining segmentation, classification, and image analysis. Neural networks, especially deep ones, significantly outperform traditional methods, providing better results in complex medical tasks. Further innovations in this area will further improve the accuracy and efficiency of diagnostics.

1. Refat, M. A. R., Al Amin, M., Kaushal, C., Yeasmin, M. N., & Islam, M. K. (2021). A Comparative Analysis of Early Stage Diabetes Prediction using Machine Learning and Deep Learning Approach. (n.d.). IEEE Conference Publication | IEEE Xplore, 654-659. https://ieeexplore.ieee.org/abstract/document/9609364 (pp. 654-659).
2. Aswathy, M. A., & Mohan, J. (2020). Analysis of machine learning algorithms for breast cancer detection. In Handbook of Research on Applications and Implementations of Machine Learning Techniques, 1-20. https://doi.org/10.4018/978-1-5225-9902-9.ch001
3. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. Communications of the ACM, 60(6), 84–90. https://doi.org/10.1145/3065386
4. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for Large-Scale image recognition. arXiv.org. https://arxiv.org/abs/1409.1556
5. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., & Rabinovich, A. (2015). Going Deeper with Convolutions. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1-9. https://www.cv-foundation.org/openaccess/content_cvpr_2015/html/Szegedy_Going_Deeper_With_2015_CVPR_paper.html
6. Berezsky, O., Pitsun, O., Derysh, B., Pazdriy, I., Melnyk, G., & Batko,. Y. (2021) Automatic Segmentation of Immunohistochemical Images Based on U-net Architecture. IEEE 16th International Conference on Computer Sciences and Information Technologies (CSIT), 29-32. https://ieeexplore.ieee.org/document/9648669

Oleh Berezsky[1], Grygoriy Melnyk[2], Fedir Kudinov[2]
[1]*West Ukrainian National University, Ternopil, Ukraine, olber62@gmail.com*
[2]*West Ukrainian National University, Ternopil, Ukraine, grygoriy.melnyk@gmail.com*

MODELS AND SOFTWARE SYSTEM FOR SCIENTIFIC INFORMATION SEARCH

Introduction. Scientometric databases are an important component of the scientific information environment. These are repositories of scientific publications, citations, and other research data. Their purpose is to collect, organize, and analyze scientific information to measure scientific impact, identify relevant topics, and support decision-making in science [1, 2]. In this context, repositories of training image datasets play a vital role in improving the search for experimental data, enabling the development of models that can efficiently locate and classify datasets linked to specific research efforts [3].

Scientometric databases include various types of scientific information and are characterized by indexing, citations, impact factors, and other metrics. Databases and search engines differ in coverage and quality of information retrieval. Metadata in scientific databases describes articles and includes the title, authors, abstract, keywords, publication source, date, DOI, and citation information.

With the development of scientific research and the growing amount of information, the need for effective tools for navigation, analysis, and use of scientific data is becoming more and more urgent. Existing search and recommendation systems [4] use web scraping and limited document description. The purpose of this work is to develop algorithms and a system for searching and collecting scientific information. The object of research is the process of searching for information. The subject of research is algorithms and models of information search.

Scientometric databases. The main characteristics of scientometric databases are indexing, citations, and impact factors. Scientometric databases Scopus, Web of Science, IEEE Xplore, Google Scholar, Internet Archive Scholar, CORE, CiteSeerX, and Semantic Scholar provide access to full texts and full-text searches, as well as metadata.

Scientific databases offer a variety of search functions, including searching by keyword, author, title, year of publication, and more. Advanced search allows the use of logical operators and combinations of search fields for more precise results. Metadata in scientific databases includes information about the title of the work, authors, abstract, keywords, publication date, DOI, and citation information.

Information retrieval models. This work presents the development and implementation of a software system for scientific articles search. Every scientific article is a document that contains certain components, which are indexed by scientometric databases. There are three main classic models of information retrieval from documents: Boolean, vector, and probabilistic. The Boolean model is based on set theory and Boolean algebra, using logical operations AND, OR, NOT to form queries. Documents are represented as sets of keywords, and searching is performed by applying Boolean operators to the query. The model is easy to implement and allows you to accurately formulate requests. However, the model does not take into account the partial coincidence of documents with the request, so it does not provide a ranking of the results.

In the vector model, documents and requests are represented as vectors in a multidimensional space. Each keyword is associated with a weighting factor that reflects its importance in the context of the document or query. The model takes into account a partial match between the document and the request, which allows ranking of documents by degree of relevance. The vector model is more flexible compared to the Boolean model. However, the vector model assumes independence of keywords, which is not always an accurate assumption.

The probabilistic model tries to estimate the probability that the document is relevant to the user's query. It assumes that for each request there is an ideal set of documents that the user wants to retrieve. This model ensures the correct ranking of documents according to the probability of relevance. It can gradually improve search results based on user feedback. However, the model does not take into account the frequency of occurrence of terms in the document and also assumes independence of terms.

Scientific article search system. The developed system provides automated data collection, processing of user requests, and integration with APIs of popular scientific publishers. The system has three components that help implement the necessary functionality: to collect information, to update data, and to search for articles. The updating data algorithm is shown in Fig.1.

The main data of the system is stored in the MySQL database. It stores information about journals and articles and various metadata such as DOI, ISSN, and other identifiers. The structure of the database is designed in such a way as to provide quick access to the necessary data and the ability to filter it according to various criteria.

The system provides an automated collection of data on scientific articles from open sources, using APIs of publishing houses. Data is collected regularly, which allows you to keep the information up to date. System users can search for articles by keywords, publication dates, journal names, and other parameters. Search results can be exported in BibTeX format for further use in bibliography managers such as JabRef.

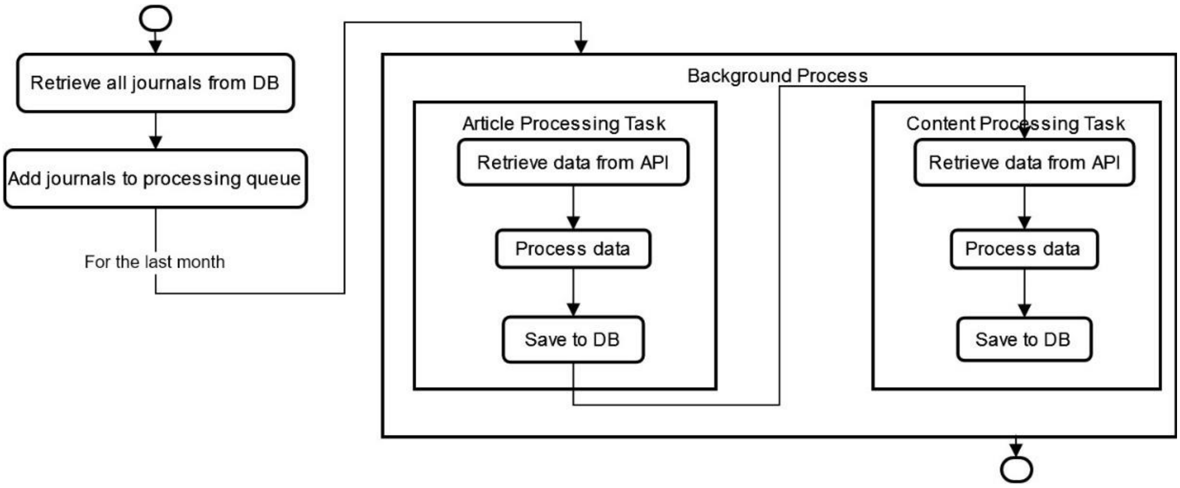


Fig. 1. Updating data algorithm

A key practical aspect of the developed system is its interaction with the APIs of Elsevier and IEEE publishers, compared to the approach described in [4].

Conclusions. The developed system demonstrates the effectiveness of the automated approach to information search in scientific databases. It provides a convenient tool for researchers to quickly find the materials they need and manage them efficiently. The use of publishing APIs ensures high performance and flexibility of the system, making it an important tool for the modern scientific community.

1. Zaragoza, H. (2002). Information Retrieval: Algorithms and Heuristics. *Information Retrieval*, *5*, 271–274.
2. Ceri, S., Bozzon, A., Brambilla, M., Della Valle, E., Fraternali, P., & Quarteroni, S. (2013). An Introduction to Information Retrieval. B S. Ceri, A. Bozzon, M. Brambilla, E. Della Valle, P. Fraternali, & S. Quarteroni (Eds.), *Web Information Retrieval* (pp. 3–11). Springer. https://doi.org/10.1007/978-3-642-39314-3
3. Berezsky, O., Liashchynskyi, P., Pitsun, O., & Melnyk, G. (2024). Method and Software Tool for Generating Artificial Databases of Biomedical Images Based on Deep Neural Networks. *CoRR*, *abs/2405.16119*. https://doi.org/10.48550/ARXIV.2405.16119
4. Putra Utama, F., Mardiansyah, T., Faurina, R., & Vatresia, A. (2023). Scientific articles recommendation system based on user's relatedness using item-based collaborative filtering method. *Jurnal Teknik Informatika (Jutif)*, *4*(3), 467–475. https://doi.org/10.52436/1.jutif.2023.4.3.702

Yurii Bilas[1], Volodymyr Riznyk[2]
[1]Bini Games, Larnaca, Cyprus, yurii.y.bilas@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, volodymyr.v.riznyk@lpnu.ua

COMPUTATIONAL INTELLIGENCE METHOD FOR EMOTIONAL GIBBERISH SPEECH SYNTHESIS

Introduction. In recent years, speech synthesis technology has advanced from robotic, unnatural sounds to realistic synthetic voices that are difficult to distinguish from human speech. Despite these improvements, text-to-speech (TTS) systems still face challenges, especially when used for non-human or artificial languages. Often, research focuses on replicating human languages like English or Ukrainian, but there are cases where generating speech in a real language is unnecessary. For example, when communicating with young children or in sound design for films and games, the emotional tone of the speech may be more important than the words themselves.

This research explores methods for creating artificial languages and preserving emotional expression in synthetic speech.

The aim is to design a system that transforms phonemes into artificial language sounds while retaining the original speaker's emotions and rhythm.

The relevance of the study lies in the fact that research and development in the field of synthesis of speech with artificial language will simplify and reduce the cost of sounding films, cartoons, games, etc.

The object of research: synthesis of speech in artificial language with emotion transfer.

The subject of research: methods and means of speech synthesis with emotions based on artificial language.

The role of artificial languages. Artificial languages are valuable in industries like entertainment and robotics. In video games like «The Sims» and «Animal Crossing» or TV shows such as «Teletubbies», artificial speech enhances immersion without needing real words. These artificial languages are often cheaper and easier to produce than real languages, eliminating the need for translation and voice actors.

Robots, such as Kismet (Breazeal, 2000) and Probo (Saldien et al., 2008), use emotional gibberish to interact with children. Artificial languages help create emotional connections across cultural and language barriers. Additionally, synthesized languages can be used in experimental music, art, and various research areas related to speech.

Analysis of the latest research and publications. Research on artificial language creation often involves manual construction by linguists or enthusiasts. For example, J.R.R. Tolkien developed languages like Quenya and Sindarin, building intricate phonetic, grammatical, and cultural systems. Such processes are time-consuming and expensive (Beinhoff, 2024). Online platforms like Langmaker.com host databases of constructed languages, but creating a full language still requires significant effort.

Recent developments in artificial intelligence, such as ChatGPT, have introduced new possibilities. In recent research, Dr. Diamond (2023) demonstrated that AI-generated languages, like those created using ChatGPT, adhere to Zipf's law, which governs word frequency distribution in natural languages. While AI can generate languages that follow linguistic laws, results are inconsistent, especially for larger texts. As an example, when we tried to create our own language via ChatGPT, we noticed that in one response, the AI translated a word as «birds», but in another sentence, the same word was translated as «mountains», indicating some instability in maintaining semantic consistency across contexts. This shows AI's potential but highlights its limitations in complex language creation.

Algorithmic approaches exist, too. Yilmazilidiz and colleagues (2015) categorize gibberish generation into methods like "jabberwocky" (nonsensical words), random splicing of audio, and syllable-based constructions. These methods either lack realism, require manual filtering, or produce results that are still recognizable as originating from real languages. Markov chain models offer an alternative, generating more varied speech patterns based on statistical analysis of phonemes.

The analysis of emotional speech synthesis highlights methods that treat emotions as parameter sets (Schröder, 2001), adjusting features like pitch, tempo, and loudness to convey different emotions. Systems using Hidden Markov Models (HMM) are often employed for this purpose (Liu et al., 2021). In some cases, emotional transfer occurs post-synthesis through prosodic transfer, as demonstrated by the ZEST system (Dutta et al., 2024), which combines elements from different recordings to apply new emotions. Ubisoft's Daft-Exprt model stands out as a breakthrough, allowing prosody and emotions to be transferred from one speaker to another while maintaining high quality (Zaïdi et al., 2022).

System design for gibberish synthesis. After reviewing popular methods in speech synthesis, we propose a solution to the problem of synthesizing artificial languages with emotional expression. The key approach is using IPA phonemes to represent sounds that are uncommon in English but present in other languages. Markov chains are going to generate gibberish by modeling speech patterns based on phonetic data from multiple languages. A good example of the data source is the «ipa-dict» corpus, which contains transcriptions in IPA for over 30 languages.

Allosaurus is going to be used to extract phonemes from input audio.

To enhance realism, the generated gibberish will follow Zipf's law, which dictates word frequency distribution in natural languages. This can be achieved by using hashing (e.g., MD5) for words and passing the hashes to a pseudo-random number generator to work with the Markov model. Markov chains will generate syllables individually, preserving the structure of the original speech to make transferring emotions easier.

For speech synthesis, we suggest using the Tacotron 2 model combined with HMM (proposed by Mehta et al., 2022) to convert phonemes into speech. For emotion transfer, we propose using the Daft-Exprt method from La Forge, which efficiently transfers emotions, rhythm, and prosody from the original speaker to the synthesized speech.

In summary, we propose a system diagram (Fig. 1) designed to solve the problem of synthesizing artificial language speech with emotional transfer. The diagram uses rounded rectangles to represent external input and output data, parallelograms for internal data generated within the system, rectangles for software components, and arrows to show the flow of data.

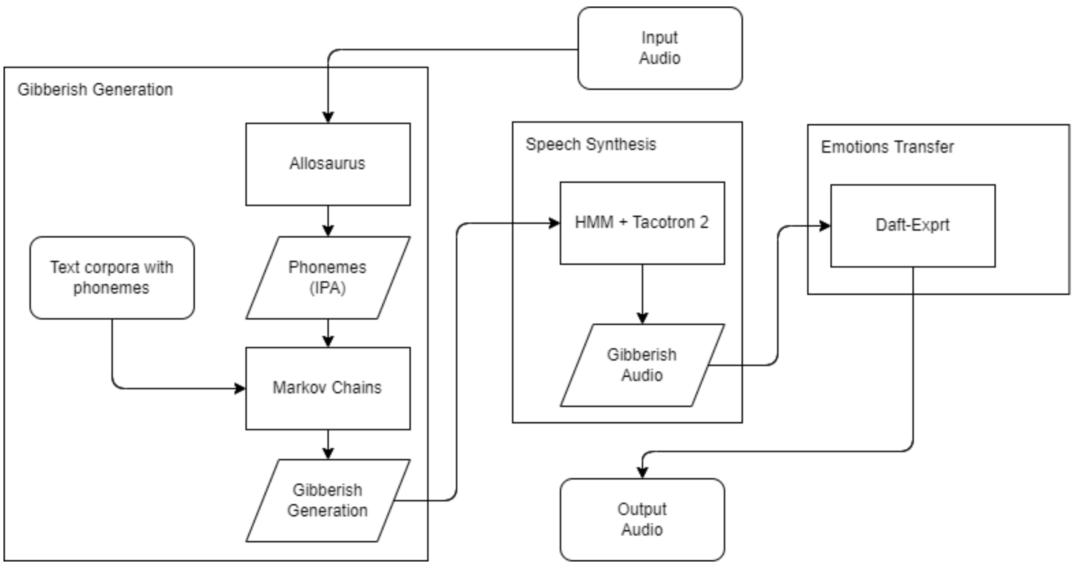


Fig. 1. Scheme of the emotional gibberish speech synthesis system

Conclusions. Existing methods for creating artificial languages, speech synthesis, and emotional transfer were analyzed. The proposed system efficiently addresses these tasks, making it useful for streamlining voiceover production in films, cartoons, and games.

The proposed system integrates components that work faster than in real-time (this means that during 1 second they're able to process more than 1 second of audio data), ensuring high-speed performance. The novelty lies in the end-to-end approach for synthesizing artificial language speech with emotional transfer, presented for the first time. The system's structure is designed to deliver both high-speed and high-quality sound.

Practical value lies in the fact that the proposed system can simplify and reduce the cost of voicing movies, cartoons, games etc. Also, it may be useful for voicing avatars, robots in healthcare, rehabilitation, child care etc.

Future work will focus on system implementation and testing with real listeners.

1. Breazeal, C. (2000). *Sociable Machines: Expressive Social Exchange Between Humans and Robots* (Sc.D. Thesis). MIT Department of Electrical Engineering and Computer Science, Cambridge, MA.
2. Saldien, J., Goris, K., Yilmazyildiz, S., Verhelst, W., & Lefeber, D. (2008). On the Design of the Huggable Robot Probo. *Journal of Physical Agents, 2*(2). http://dx.doi.org/10.14198/JoPha.2008.2.2.02
3. Beinhoff, B. (2024). Design intentions and actual perception of fictional languages: Quenya, Sindarin, and Na'vi. In I. A. C. NOLETTO, J. NORLEDGE, & P. STOCKWELL (Eds.), Reading Fictional Languages (pp. 76–92). Edinburgh University Press. http://www.jstor.org/stable/10.3366/jj.9941158.10
4. Diamond, J. (2023). "Genlangs" and Zipf's Law: Do languages generated by ChatGPT statistically look human? *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2304.12191
5. Yilmazyildiz, S., Read, R., Belpeame, T., & Verhelst, W. (2015). Review of Semantic-Free Utterances in Social Human–Robot *Interaction. International Journal of Human-computer Interaction*, *32*(1), 63–85. https://doi.org/10.1080/10447318.2015.1093856
6. Schröder, M. (2001). Emotional speech synthesis: a review. *Eurospeech 2001 Scandinavia, 7th European Conference on Speech Communication and Technology, 2nd Interspeech Event*. http://doi.org/10.21437/Eurospeech.2001-150
7. Liu, R., Şişman, B., & Li, H. (2021). Reinforcement Learning for Emotional Text-to-Speech Synthesis with Improved Emotion Discriminability. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2104.01408
8. Dutta, S., & Ganapathy, S. (2024). Zero shot audio to audio emotion transfer with speaker disentanglement. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2401.04511
9. Zaïdi, J., Seuté, H., Van Niekerk, B., & Carbonneau, M. (2022). Daft-Exprt: Cross-Speaker Prosody transfer on any text for expressive speech synthesis. *Interspeech 2022*. https://doi.org/10.21437/interspeech.2022-10761
10. Mehta, S., Székely, È., Beskow, J., & Henter, G. E. (2022). Neural HMMS are all you need (For High-Quality Attention-Free TTS). *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. https://doi.org/10.1109/icassp43922.2022.9746686

Bohdan Borkivskyi[1], Vasyl Teslyuk[2]
[1]*Lviv Polytechnic National University, Lviv, Ukraine, bohdan.p.borkivskyi@lpnu.ua*
[2]*Lviv Polytechnic National University, Lviv, Ukraine, vasyl.m.teslyuk@lpnu.ua*

OBJECT DETECTION OPTIMIZATION IN CLOSED SPACE USING MOBILE ROBOTIC SYSTEMS WITH OBSTACLE AVOIDANCE

Introduction. In recent years, the introduction of mobile robotic systems has been observed in various fields, ranging from industrial automation to service robotics and autonomous vehicles. Central to the functioning of such systems is their ability to perceive and interact with the environment [1], which is often accompanied by the need to detect objects. Traditional methods of object detection typically involve defining bounding boxes around objects of interest in photo or video frames. While these methods are effective in many scenarios, they often have trouble accurately defining object boundaries and handling failures, which are common problems in real-world environments. As a result, there is a growing interest in using object segmentation methods to improve the efficiency of object detection by mobile robots.

The importance of optimizing object detection for mobile robots [2] is that it directly affects the efficiency, reliability, and safety of robotic systems operating in dynamic and unstructured environments. In industrial environments, accurate object detection is crucial for tasks such as automated material handling, assembly, and quality control, where errors or inconsistencies can lead to significant disruptions and economic losses. In service robotics, including home care, healthcare, and retail, reliable object detection enables robots to efficiently assist humans and confidently navigate complex environments. In addition, in autonomous vehicles and unmanned aerial vehicles (UAVs), accurate object detection is essential for collision avoidance, route planning, and situational awareness, ensuring safe and efficient operation in a variety of scenarios.

Despite advances in object detection methods, challenges remain in optimizing the performance of mobile robots, especially in terms of accuracy, speed, and resource efficiency. Traditional object detection methods often have difficulty accurately detecting small or enclosed objects, distinguishing between similar objects, and adapting to changes in lighting conditions or environmental factors. By using object segmentation methods and optimizing their integration with object detection pipelines, researchers and practitioners are striving to solve these problems and unlock the full potential of mobile robots in various fields. Accordingly, the optimization of obstacle recognition in mobile systems is an urgent task of today.

Materials and methods. The presented method is a combination of a deep neural network-based approach for object boundary recognition and visual data for obstacles in a closed space. The base for the object detection neural network is a Deeplab model architecture [3], trained on the NYU Depth Dataset V2 [4] and ADE20K [5] datasets with extensive data for varying scene types and object categories along with corresponding annotations. The paper provides the results of several conducted experiments aiming to estimate the performance and feasibility of using the developed system for various tasks with the UNet [6] and Deeplab architectures on different datasets. The experiments determined that the developed system built utilizing the Deeplab neural network architecture and the ADE20K scene parsing dataset reached an accuracy of 86.9% in multi-object segmentation. The visual results were presented to demonstrate the obstacle detection of various objects in a closed space in different situations, like an empty apartment with random obstacles on the way or a crowded study space at a university. The results show an excellent distinction between room sides like ceiling and floor, walls and doors, as well as detecting people and pieces of furniture.

Results. Thus, based on the results of the work performed, the following scientific novelty and practical significance of the research results can be formulated.

The scientific novelty of the research results is that the method of object recognition using neural networks has been improved by modifying the internal structure of the neural network to obtain accurate boundaries of objects in the image.

The practical significance of the research results is that the improved method of object recognition can be used in military robotic systems in various conditions, for example, during demining to detect dangerous objects or during rescue and search operations, as well as in robotic systems for social purposes, for example, in robot assistants.

Conclusions. As a result of the work performed, the existing software solutions for solving the problem of finding the exact boundaries of an object in an image were investigated, and existing approaches to the design and construction of neural networks for object boundary recognition were analyzed.

To ensure the quality of the model, the NYU Depth Dataset and ADE20K datasets, which contain a large number of images of the closed environment, were chosen for training. The method of object recognition using neural networks has been improved by modifying the internal structure of the neural network to obtain accurate boundaries of objects in the image, namely by adding atroscopic layers and applying a change in data dimensionality to detect hidden properties.

To determine the efficiency of the system, experiments were conducted on the NYU Depth Dataset and ADE20K datasets, and the object recognition accuracy was achieved at 87%. The system is able to recognize typical objects and objects found in an indoor environment, such as walls, ceilings, floors, furniture, foreign objects, etc.

An example of a situation where a solution can be useful is in the area of auxiliary robots. Such a system can be an assistant, helping a person navigate through a building and help them find the things they need. Also, the developed system can perform tasks and work autonomously, move independently in a closed environment, and automatically avoid obstacles ahead.

1. Tsmots, I. G., Teslyuk, V. M., Opotiak, Yu. V., & Oliinyk, O. O. (2023). Development of the scheme and improvement of the motion control method of a group of mobile robotic platforms. Ukrainian Journal of Information Technology, 5(2), 97–104. https://doi.org/10.23939/ujit2023.02.097
2. Borkivskyi, B. P., Teslyuk, V. M. (2023). Application of neural network tools for object recognition in mobile systems with obstacle avoidance. Scientific Bulletin of UNFU, 33(4), 84-89. https://doi.org/10.36930/40330412
3. Vinh Quoc Luu, Le, D. K., Nguyen, H. T., Nguyen, M. T., Nguyen, T. T., & Dinh, V. Q. (2024). Semi-Supervised Semantic Segmentation using Redesigned Self-Training for White Blood Cells. arXiv. https://doi.org/10.48550/arXiv.2401.07278
4. Silberman, N., Hoiem, D., Kohli, P., & Fergus, R. (2012). Indoor Segmentation and Support Inference from RGBD Images. In Computer vision – ECCV 2012 (pp. 746–760). Springer. https://doi.org/10.1007/978-3-642-33715-4_54
5. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., & Torralba, A. (2017). Scene Parsing through ADE20K Dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 5122–5130). https://doi.org/10.1109/CVPR.2017.544
6. Liu, G., Guo, Y., Wu, C., Chen, G., Saheya, B., & Jin, Q. (2023). A Region of Interest Focused Triple UNet Architecture for Skin Lesion Segmentation. arXiv. https://doi.org/10.48550/arXiv.2311.12581.

Volodymyr Chornenkyi[1], Iryna Kazymyra[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, volodymyr.y.chornenkyi@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, iryna.y.kazymyra@lpnu.ua

INFLUENCE OF THE MACHINE LEARNING MODEL CHARACTERISTICS ON PERFORMANCE IN MOBILE APPS

Introduction. Mobile devices, encompassing smartphones and tablets, have become ubiquitous in modern life. According to the International Telecommunication Union (ITU), there were over 5.3 billion unique mobile phone users globally by the end of 2022, reflecting an unprecedented level of connectivity and engagement with mobile technology [1]. The widespread adoption of mobile devices is driven by their affordability, accessibility, and the extensive range of functionalities they offer. Mobile devices are now integral to everyday activities, from communication and entertainment to productivity and health management [2].

Machine learning (ML) has increasingly been integrated into mobile applications, enabling sophisticated features and services directly on users' devices. The proliferation of on-device ML is largely attributed to advancements in mobile hardware and software frameworks designed to support such technologies. For instance, Google's TensorFlow Lite and Apple's Core ML provide specialized tools for deploying ML models efficiently on mobile platforms.

The application of ML on mobile devices ranges from real-time image and speech recognition to predictive text input and personalized recommendations [3-4]. Research indicates that on-device ML can significantly enhance user experience by providing faster responses and preserving privacy through local data processing [5]. As of 2023, an increasing number of mobile applications are leveraging ML for functionalities like augmented reality (AR), health monitoring, and autonomous navigation [6].

This paper explores the influence of machine learning model characteristics – specifically Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs) – on performance in mobile environments. It also examines the impact of optimization techniques such as quantization on these models. By understanding these factors, we aim to provide insights into how different ML architectures and optimizations can be effectively utilized for mobile applications.

Materials and methods. For experiment purposes, models with 2 different architectures and 2 datasets were used (Table 1). As CNN architecture representative, MobileNetv2 was used. MobileNet v2 is a lightweight deep learning model developed by Google, specifically designed for mobile and edge devices. It employs depthwise separable convolutions, which break down standard convolution operations into simpler components, significantly reducing the number of parameters and computational complexity. The architecture features linear bottlenecks, where the last layer of each block is a linear layer instead of a non-linear activation function, enhancing information preservation and gradient flow during training. Additionally, MobileNet v2 incorporates residual connections to facilitate the learning of identity mappings, helping to mitigate the vanishing gradient problem and improve feature extraction [7]. As for ViT architecture, a default structure was used. The core of the architecture consists of multiple transformer encoder layers, each featuring a self-attention mechanism that allows the model to assess the importance of different patches relative to one another. Following the self-attention layer, the output is processed through a feed-forward neural network, with normalization and residual connections enhancing training stability and performance. A special classification token is included in the sequence of patch embeddings, aggregating information for the final classification task. The output corresponding to this classification token is then passed through a linear layer to produce class probabilities. Overall, ViT represents a significant shift from traditional convolutional neural networks (CNNs) by treating images as sequences of patches, leveraging self-attention mechanisms to achieve competitive performance across various image classification benchmarks while offering a flexible and scalable architecture [8].

Additionally, models were also passed through quantization optimization which should provide significant advantages for mobile use by reducing model size, speeding up inference, lowering power consumption, and improving hardware utilization, all while maintaining acceptable accuracy levels.

For variability purposes, ImageNet and ASL Alphabet datasets were used. ImageNet dataset that includes 1000 labels and contains 1,281,167 training images, 50,000 validation images, and 100,000 test images. Image size is 224x224 pixels [9]. In comparison, the ASL Alphabet dataset contains 87,000 images, 200x200 pixels in size, and 29 classes [10]. This should beneficially influence the model size and simulate usage for sign language recognition use cases.

Table 1. Models and datasets used for experiments

Model	Architecture	Dataset
MobileNet v2	CNN	ImageNet
Quantized MobileNet v2	CNN	ImageNet
MobileNet v2	CNN	ASL Alphabet
MobTransformer	ViT	ASL Alphabet
Quantized MobTransformer	ViT	ASL Alphabet

Model size is one of the most critical characteristics of ML models for mobile devices. Mobile devices typically have limited computational resources compared to desktop or server environments. Therefore, models must be lightweight to ensure they can be deployed effectively without consuming excessive memory or processing power. Smaller models not only save memory but also reduce latency or inference speed, which is crucial for real-time applications such as image recognition or natural language processing. The inference speed can be influenced by various factors, including the model architecture, the optimization techniques used (such as quantization and pruning), and the underlying hardware capabilities.

Results. To better understand which models suit our specific needs we evaluated a number of parameters important for the usage on mobile phones: model physical size, random-access memory (RAM) and central processing unit (CPU) usages, and inference time.

All experiments were conducted on an Android device – Xiaomi Redmi Note 8 Pro equipped with Mediatek MT6785 CPU (8 cores), and 6Gb of RAM. Metrics were measured with built-in profilers in an Android Studio integrated development environment over 1200 iterations. Integration of the models into the mobile app was implemented using TensorFlow Lite without graphics processing unit usage [11].

Table 2. Results of the experiments

Model	Physical size	RAM usage	CPU usage	Inference time (percentile / ms)
MobileNet v2 + ImageNet	14 Mb	140 – 155 Mb	18 – 22%	0,95 / 53ms 0,9 / 46ms 0,5 / 37ms 0,1 / 32ms
Quantized MobileNet v2 + ImageNet	3.6 Mb	140 – 155 Mb	18 – 20%	0,95 / 65ms 0,9 / 58ms 0,5 / 48ms 0,1 / 47ms
MobileNet v2 + ASL Alphabet	3.3 Mb	120 – 135 Mb	12 – 16%	0,95 / 50ms 0,9 / 46ms 0,5 / 36ms 0,1 / 33ms
ViT + ASL Alphabet	3.5 Mb	118 – 145 Mb	7 – 14%	0,95 / 16ms 0,9 / 16ms 0,5 / 12ms 0,1 / 7ms
Quantized ViT + ASL Alphabet	899 Kb	109 – 140 Mb	7 – 9%	0,95 / 10ms 0,9 / 10ms 0,5 / 5ms 0,1 / 3ms

Conclusions. The results (Table 2) indicate that quantization can lead to significant reductions in model size, making it more suitable for mobile deployment. While there may be some trade-offs in inference time for certain models, the overall efficiency in terms of CPU usage and the drastic reduction in physical size make quantized models a compelling choice for mobile applications. The performance of the models also highlights the importance of dataset selection, as it can impact both accuracy and inference speed. Ultimately, the choice between quantized and non-quantized models should consider the specific application requirements, including the need for speed, resource constraints, and the nature of the data being processed.

To discover further areas for improvement in machine learning models for mobile deployment, several experiments can be conducted. These include comparing various quantization techniques, exploring alternative lightweight architectures, and optimizing hyperparameters to assess their impact on performance. Evaluating models on diverse datasets and under real-world conditions can provide insights into robustness and generalization.

1. ITU/UN tech agency. (2024, May 19). Measuring digital development - Facts and figures 2023. ITU. https://www.itu.int/hub/publication/d-ind-ict_mdd-2023-1/
2. Statista. (2024, June 14). Topic: US smartphone market. https://www.statista.com/topics/2711/us-smartphone-market/#topicOverview
3. Fedasyuk, D. V., & Kostiuk, M. O. (2024). Forecasting of soil moisture using machine learning in smart agriculture systems. Ukrainian Journal of Information Technology, 6(1), 26–36.
4. Teslyuk, V. M., & Ivasiv, S. S. (2024). Models and means of clothing elements patterns classification using machine learning. Ukrainian Journal of Information Technology, 6(1), 37–47. https://doi.org/10.23939/ujit2024.01.037
5. Brand, L., Humm, B. G., Krajewski, A., & Zender, A. (2023). Towards improved user experience for artificial intelligence systems. In *Communications in computer and information science* (pp. 33–44). https://doi.org/10.1007/978-3-031-34204-2_4
6. Li, Y., Dang, X., Tian, H., Sun, T., Wang, Z., Ma, L., Klein, J., & Bissyande, T. F. (2022). AI-driven Mobile Apps: an Explorative Study. *arXiv*. https://doi.org/10.48550/arxiv.2212.01635
7. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. *arXiv*. https://doi.org/10.48550/arxiv.1801.04381
8. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2020). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv*. https://doi.org/10.48550/arxiv.2010.11929
9. Deng, J., Dong, W., Socher, R., Li, L., Li, N. K., & Fei-Fei, N. L. (2009). ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 248–255). https://doi.org/10.1109/cvpr.2009.5206848
10. Kaggle. (2018, April 22). ASL alphabet. https://www.kaggle.com/datasets/grassknoted/asl-alphabet
11. TensorFlow. (2024, July 28). TensorFlow lite for Android. https://www.tensorflow.org/lite/android

Ivan Danych[1], Zoreslava Shpak[2]
[1] Lviv Polytechnic National University, Lviv, Ukraine, dynychivan@gmail.com
[2] Lviv Polytechnic National University, Lviv, Ukraine, zoreslava.y.shpak@lpnu.ua

PRACTICAL AND EDUCATIONAL APPLICATION FOR INTERACTION WITH GRAPHS

Introduction. Graph theory is an important field of mathematics and computer science that has widespread usage in various areas – from computer science to smart systems and social networks. However, the complexity of this theory can make a barrier to its understanding and comprehensive usage. At the same time learning the material is usually easier when information is presented visually. In the case of graph theory this is especially important because it is much easier to analyze graphs in a graphical format than in other forms. Therefore, tools for graph visualization are in high demand.

There are some ready-made solutions for visualizing graphs that use different algorithms with their own unique features, allowing exploring the same graph from different perspectives. However, most of these tools are aimed only at graph visualization and do not provide the ability to interact with the graph or to analyze graph theory algorithms and the processes for solving graph problems. The diversity of graph theory algorithms can lead to difficulty in choosing the optimal approaches. Moreover, users need to understand which algorithms are best suited for specific scenarios. Also often there is a need to adapt known graph algorithms to particular conditions or real-world tasks.

Methods and results. The developed and presented software application is designed for the visualization of algorithms and processes for solving graph theory problems. Its toolkit includes convenient and intuitive tools for graph input, representation, and research. Our application allows users to implement basic graph problem-solving algorithms and supports the ability to configure these algorithms. All this not only facilitates understanding and deeper mastery of graph theory but also provides users with the ability to explore different approaches and solution methods.

In the process of designing and developing the application, technologies such as Java, JavaFX, Spring Core, GraphStream, and others were used. In particular, JavaFX, an open-source library for developing graphical interfaces in the Java programming language, was used to design the application's graphical interface. JavaFX stands out for its high functionality and wide range of features compared to other similar tools.

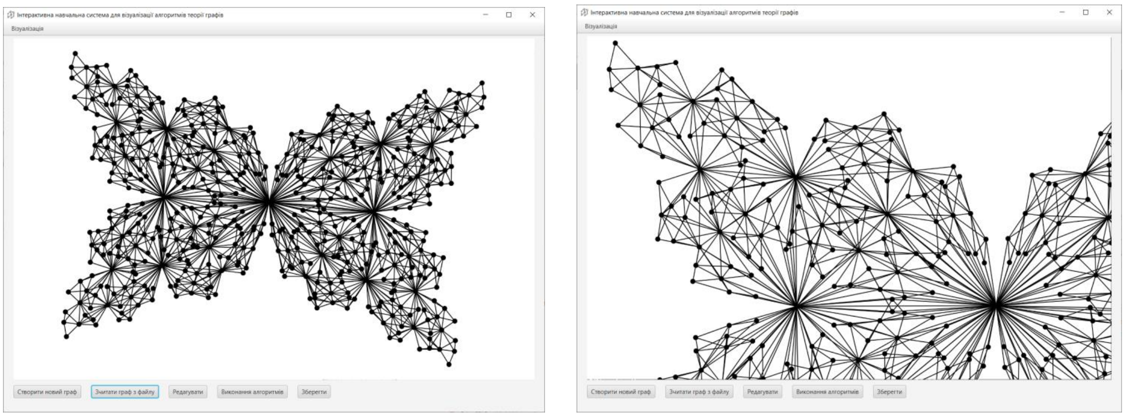
The Spring Core library greatly simplifies the process of configuring and creating service objects in software products. It offers the use of an object container that automatically creates and configures them, providing the necessary dependencies for each object. GraphStream is another powerful library designed for working with and visualizing graphs in Java. It provides developers with convenient tools for creating, modeling, and displaying graphs in applications.

Force-Directed Layout, namely its special case Fruchterman-Reingold [1], was chosen as the basic visualization algorithm. This algorithm gradually adapts the positioning of the graph's vertices on the image by considering simulated physical interaction forces between them. Step by step, it improves the clarity and aesthetic appearance of the graph. The dynamic updating and modification capability is a significant advantage. It allows efficient adaptation to changes in the graph, ensuring continuous updating of vertex positions when elements of the graph are added or removed.

A further development of the Fruchtermann-Reingold algorithm is its modification that uses well-separated pairwise decomposition (WSPD) [2]. This improved method increases computational efficiency by reducing the number of interactions between nodes, which is especially useful for large graphs.

The functionality of the developed application includes tools for creating and editing graphs, options for reading graphs from a file and writing data to a file, visual representation of graphs, tools for editing graphs and managing visualization, software implementations of basic graph problem-solving algorithms with the ability to configure them, as well as step-by-step execution and observation of the problem-solving processes.

One of the main features is graph visualization, which allows displaying graphs with even a large number of vertices and edges (Fig. 1a). Since vertex labels in images of large graphs can interfere with their visual clarity, the option to disable label display is provided. Additionally, an important functionality is the ability to change the scale (zoom in or out) of the graphical representation of the graph, ensuring the necessary apparency of the image (Fig. 1b).



(a) *(b)*

Fig. 1. Example of visualizing a large graph (a) and image scaling capabilities (b).

An important feature of the developed application is the ability to observe an algorithm step-by-step execution process with the display of main current step results on the graph. The user can move to the next step or return to the previous one. This allows analyzing the progress of the algorithm in detail. Fig. 2 illustrates the display of the first two steps in the implementation of the well-known Dijkstra's algorithm.

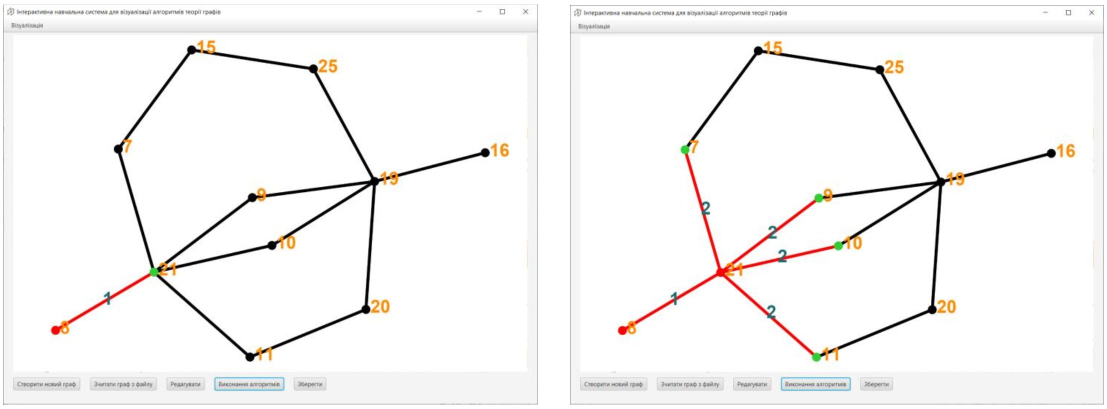
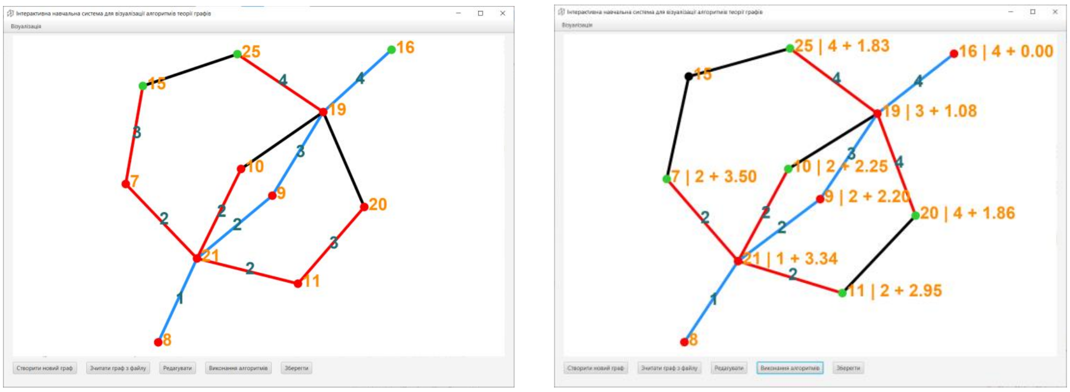


Fig. 2. Display of the first two steps of Dijkstra's algorithm execution.

In the image above, the traversed vertices and edges are marked in red, and each processed edge displays the cumulative transition cost from the starting vertex (8), taking into account the already analyzed path (the graph in Fig. 2 is unweighted, so the weight of each edge is 1). The user can change the colors and sizes of the labels, vertices, and edges to ensure maximum visualization clarity.

The practical value of the application is also based on the ability to compare the efficiency of different algorithms, especially those that solve the same problem or have a common goal. For example, to designate the shortest path between two given vertices of a graph, in addition to Dijkstra's algorithm, the A* algorithm can also be used. Fig. 3 shows the results of the implementation of these algorithms for the same graph.



(a) *(b)*

Fig. 3. The shortest path between vertices 8 and 16:
(a) – Dijkstra's algorithm; (b) – A* algorithm.

Since Dijkstra's algorithm is based on breadth-first search, it often processes many "unnecessary" vertices that will not be included in the final path, especially when the shortest path spans the entire graph. The A* algorithm, on the other hand, uses a heuristic function to estimate the distance to the target vertex, allowing the search to be directed towards the goal. As seen in Fig. 3b, some vertices remain black (unvisited), while others turn green (opened vertices that have not been processed). Comparison of the results of these two algorithms shows that A* algorithm is more efficient, as it requires significantly fewer steps.

The application supports step-by-step execution of other algorithms, such as Floyd-Warshall, Kruskal, and Prim, as well. The Floyd-Warshall algorithm, used for the computation of the shortest paths between all pairs of vertices, demonstrates an iterative process of gradually improving the result. For Kruskal and Prim algorithms, which are used to build the minimum spanning tree of a graph, users can control the step-by-step construction of this tree.

Conclusions. The developed application can be used both for educational purposes to visually familiarize students with basic concepts and methods of graph theory, as well as for solving many practical problems involving graphs. The ability to easily and quickly create or modify the graphical representation of a graph, explore the characteristics of graph algorithms in various scenarios, compare their performance, and optimize them for specific tasks – all of this can be valuable for a wide range of users.

1. Fruchterman, T. M. J., & Reingold, E. M. (1991). Graph drawing by force-directed placement. Software Practice and Experience, 21(11), 1129–1164. https://doi.org/10.1002/spe.4380211102
2. Lipp, F., Wolff, A., & Zink, J. (2016). Faster Force-Directed Graph Drawing with the Well-Separated Pair Decomposition. Algorithms, 9(3), 53. https://doi.org/10.3390/a9030053

Emil Faure[1], Andrii Nykonenko[2]
[1]Cherkasy State Technological University, Cherkasy, Ukraine, e.faure@chdtu.edu.ua
[2]Cherkasy State Technological University, Cherkasy, Ukraine, andrey.nikonenko@gmail.com

NOISE-TO-TEXT METHOD ANALYSIS FOR EVALUATING AI-GENERATED TEXTS

Introduction. Rapid advancements in the automatic text generation field are already significantly influencing human life with a major impact on education, work, and personal life. There is no doubt that current progress in generative AI could bring a lot of help to different areas of human activities, processing ways to automate a lot of day-to-day activities and repetitive tasks. As with any other technology, there are some risks tightly coupled with new opportunities that the advancement of LLMs brings to us; among the most obvious risks are fake news generation, the spread of misinformation, academic misconduct, and some others. Regulation of AI, ethical guidelines, and development of best practices could help here, but they will most probably not solve the whole problem. The wide availability of AI Detection tools is the necessary prerequisite for a broader safe adoption of AI in everyday life. Despite the significant amount of such tools and their generally high performance, there is a missing piece. This piece is a complete absence of understanding of the underlying principles that allow this type of tool to operate effectively. The current state in this area could be described as “one black box (AI Detector) analyses results of another black box (LLM)”, which creates an absence of trust among AI Detection tools users. The principal absence of any additional evidence on top of the decision made by a detector makes it impossible for a human to double-check a received verdict.

Relevance of the research. Because of a lack of trust and evidence, a lot of instructors prefer to avoid using AI Detection tools in their day-to-day work, which increases the risk of academic misconduct among their students. One possible way to address this state of things is to make AI Detection technology interpretable and understandable for end users. The key thing that needs to be defined to cover the existing gap is what text characteristics make text AI-generated and what makes it human-written. There should be some trace left by an LLM during text generation, which is visible to AI Detection tools and not visible to a human being; we call that trace AI footprint. This research is focused on revealing the nature of AI footprint using specially created methods to disclose it.

Scientific novelty. AI Writing Detection is a hot research topic nowadays, and hundreds to thousands of works have been published in recent years in this area. Typically, research papers focus on specific directions: (1) LLM-generated data quality analyses and NN (neural network) architecture choices, amount of parameters, and specifics of training techniques and data flow inside the NN. The most fundamental works here are (Ouyang et al., 2022) and (Meta, 2023). (2) Research focused on the development of AI Detection models. This is a broad topic with a lot of competing approaches, where each approach has its own strengths and weaknesses. Articles (Crothers, Japkowicz, & Viktor, 2023), (Gehrmann, Strobelt, & Rush, 2019) and (Yan et al., 2023) provide an in-depth discussion of the related problems together with a deep analysis. (3) Studies focused on the evaluation of existing AI detection software, starting from massive evaluations that include a comparison of the performance of more than a dozen detectors (Walters, 2023) and to specific edge cases like the effect of the different text modification attacks on detectors (Ghosal et al., 2023).

A large amount of ongoing research allows both AI generation and detection areas to repeatedly evolve, adapting to changes and covering gaps spotted by researchers. But there is one gap which has not been addressed till now, and this gap is the interpretability of AI Detector predictions. Our study addresses the missing piece - we are trying to define what the AI footprint is and how it could provide us with human-understandable evidence. To discover the nature of the AI footprint, we propose a novel method of noise transformation across multiple stages, generating a sequence of data samples that range from completely nonsensical to fully coherent.

Most of the existing research in the AI Detection area is focused on the analysis of detection quality in some specific narrow cases or talks about ways to improve detection performance and efficiency when our method is focused on building the understanding of core principles of detection via applied research. Creating multiple versions of the same text that are pretty close between neighbour stages but are very distant for further stages, we receive a chain of coherent data with an increasing amount of AI fraction in it. Passing this data through a set of AI Detectors allows us to numerically measure their reaction to different fractions of AI in text. During the transformation stages, we are increasing the meaningfulness of the text measuring the impact on the AI footprint by detectors. This allows us to make some conclusions regarding the relation between text meaningfulness and AI footprint nature.

Highlighting the existing connection between text meaningfulness and AI detection efficacy, our research proposes a new perspective for the AI detection field. It emphasises the need for a deeper investigation of key principles of work of AI detection software and the importance of a broader analysis of AI footprint nature. This knowledge is essential for creating the next generation of more reliable, interpretable, and foolproof AI detection tools, which should help us to address key existing challenges related to LLMs, including academic misconduct and fake information, and to build better guidelines in the ethical use of AI.

Methods and materials. In our research, we propose a new way of exploring the AI footprint with the use of a noise-to-text approach. This approach describes an iterative process for generating textual data using LLM. At the very beginning, we start by generating a random noise and then consistently improve the data quality from stage to stage. At all stages of the process, we use only LLM-generated data without any direct human intervention. Our generated data mimics normal text structure to be used as any other type of text in the AI Detectors analysis. The main goal of such research is to define a text meaningfulness measure, after which the AI footprint becomes visible for the detection software.

Our experimental approach consists of 6 main stages, where we apply a very small text quality improvement between stages. This allows us to analyse samples of the same (in the ideal case) or at least close text on different meaningfulness levels. The full description is provided in the Extended Version of the paper. Data generated at each stage was then analysed using five different AI detectors (Turnitin, GPTZero, Originality, Scribbr, and Quillbot) to receive AI scores for further comparison.

- Stage 1. Completely random generation.
- Stage 2. At this stage, we are starting to add first meaningful information.
- Stage 3. The next step is to slightly increase the amount of meaningful words in each sentence.
- Stage 4. The whole dictionary is replaced, there shouldn't be any meaningless words in the text.
- Stage 5. Sentences are coherent and meaningful, on a paragraph level text stays meaningless.
- Stage 6. All sentences in a paragraph are connected and coherent, but paragraphs stay unconnected.

Results of Research. All the created data were passed through 5 selected detectors, Table 1 presents the final aggregated statistics, detailed results can be found in the Extended Version of the paper.

Table 1. Average AI Writing detection score for 10 texts per detector per stage

	Turnitin	GPTZero	Originality	Scribbr	Quillbot	Stage average
Stage 1	0%	4%	3%	5%	0%	2.4%
Stage 2	0%	4%	7%	13%	0%	4.8%
Stage 3	0%	5%	11%	8%	1%	5.0%
Stage 4	0%	6%	19%	7%	5%	7.4%
Stage 5	0%	50%	99%	22%	29%	40.0%
Stage 6	75%	87%	100%	87%	81%	86.0%

Our experimental data analysis showed a significant difference between AI scores for texts with various degrees of meaningfulness. The tendency persists despite different sensitivity levels between different tools. Some detectors, like Originality, showed a very high degree of sensitivity, while others were less sensitive. The experiment showed that with the increase in the meaningfulness of the data, AI scores increased among all the detectors. This rise indicates that AI detectors are more capable of detecting AI-generated text, which is more coherent and better structured.

The found trend shows that AI detection tools could rely mostly on the detection of templates mostly related to the way in which LLMs typically express their thoughts rather than some specific AI-generated tokens, word choice, or lexical specifics. The fact that data generated in Stages 1-3, which mostly consist of random noise and meaningless content is almost indistinguishable from human writing from the detector's standpoint highlights this point. It shows that detectors are mostly tuned to well-organised logical chains of thoughts typical for AI writing rather than to other less meaningful and more random AI-generated examples. This finding is crucial as it emphasises the limitation of state-of-the-art detection systems: they could be ineffective against AI content that lacks typical coherence, this could be used as a detection avoidance technique. A better understanding of the limitations of existing technology is crucial for the creation of guiding principles for the next generation of AI Detectors.

Practical value. This research presents valuable information obtained during an applied application of the proposed method about the potential nature of the AI footprint. Received knowledge is essential for the creation of a new evidence-based generation of AI detectors to cover the existent technological gap in the area and support the rise of trust between AI detection tools and their early adopters and educators. A deeper understanding of how AI detection relies on templates allows us to rethink our perception of AI detection technology and find a better, more reliable way of using it. The study also lays the foundation for further research regarding evidence-based AI detection.

Conclusions. This research proposes a novel approach for AI footprint identification using the noise-to-text method. Practical results received in the study could bring some light onto the guiding principles of AI Detection tools, but further research is needed to confirm whether the patterns found in the research are indeed a trace of AI or some other type of artefacts. More research in this area is crucial for the detector's reliability and interpretability improvements, which should help with building a better understanding and bringing more trust to automated detection.

1. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., ... & Lowe, R. (2022). Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, *35*, 27730-27744.
2. Meta, A. I. (2023). Introducing LLaMA: A foundational, 65-billion-parameter large language model. *Meta AI*.
3. Crothers, E. N., Japkowicz, N., & Viktor, H. L. (2023). Machine-generated text: A comprehensive survey of threat models and detection methods. *IEEE Access*, *11*, 70977-71002.
4. Gehrmann, S., Strobelt, H., & Rush, A. M. (2019). Gltr: Statistical detection and visualization of generated text. *arXiv preprint arXiv:1906.04043*.
5. Yan, D., Fauss, M., Hao, J., & Cui, W. (2023). Detection of AI-generated essays in writing assessments. *Psychological Test and Assessment Modeling*, *65*(1), 125-144.
6. Walters, W. H. (2023). The effectiveness of software designed to detect AI-generated writing: A comparison of 16 AI text detectors. *Open Information Science*, *7*(1), 20220158.
7. Ghosal, S. S., Chakraborty, S., Geiping, J., Huang, F., Manocha, D., & Bedi, A. S. (2023). Towards possibilities & impossibilities of ai-generated text detection: A survey. *arXiv preprint arXiv:2310.15264*.

Oleh Hasii[1], Mariana Seneta[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, oleh.hasii.mknus.2023@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, mariana.y.seneta@lpnu.ua

POTENTIAL OF RASPBERRY PI 5 FOR SMART SYSTEMS AND IOT: ADVANTAGES, CAPABILITIES AND COST-EFFICIENCY

Introduction. Smart systems and the Internet of Things (IoT) are becoming more and more integrated not only into our daily lives but also into different industrial and business sectors. They transform business processes, increasing productivity and dramatically changing the way we interact with technologies. With this in mind, the implementation of such technologies slowly switches from an "option" to a "requirement" for businesses to stay competitive and efficient. Considering these circumstances Raspberry Pi 5 might be a promising choice.

Raspberry Pi 5 is the latest version of a popular single-board microcomputer released in 2023. Thus, the purpose of this research is to explore the possibilities and advantages of this platform, not only as a learning tool but also in the context of enhancing the capabilities of businesses, smart systems, and IoT.

Cost-effectiveness is a critical factor for smart system integration and expansion. Most options on the market are not affordable for ordinary users and small businesses. But the Raspberry Pi 5 offers the hope of solving the issue of the balance between price and performance.

Technical specifications. The Raspberry Pi 5 is the most recent iteration of a single-board microcomputer. It is designed to connect to a variety of peripheral devices and modules. Regardless of its size, it can be considered a fully functional pocket computer. Main features and characteristics are the next:

- the core of the platform is the ARM Cortex-A76 quad-core processor, which is 2-3 times faster than its previous models
- it has 4 or 8 Gb of RAM, depending on the choice
- new GPU VideoCore VII
- support of 2 4k/60 fps displays, which creates new opportunities for applications and increases the comfort of development
- provides a connection to an external camera module
- does not have internal memory, requires an SD card to function
- has two USB 3.0 ports and 2 USB 2.0 ports
- Bluetooth 5.0 allows connection with other smart devices
- Wi-Fi operates at 2.4 GHz and 5.0 GHz. There is also a Gigabit Ethernet port.
- compact size 86×56×16 mm and weight 46 g.

This hardware platform offers quite impressive characteristics. The combination of CPU and GPU deserves special attention: a double increase in speed reduces dependence on cloud computing, which allows you to transfer some operations directly to the hardware of the system [1].

On the market, there are analogs of Raspberry Pi 5, such as ODROID-X2, BeagleBone, and Asus Tinker Board, and each of them offers similar capabilities for developing smart systems and IoT. However, Raspberry Pi 5 has several important advantages that, in my opinion, make it one of the most versatile options. For starters, RP5 has one of the largest developer communities in the single-board computer industry, providing support and access to numerous resources, documentation, and examples. This also makes RP5 more accessible on the market than, for example, less common ODROID boards. Secondly, some of the alternatives may be more focused on narrow areas. For example, BeagleBone AI 64 will be over-specialized for projects that are not centered around AI, while Asus Tinker Board is better suited for multimedia projects. In contrast, RP5 is better suited for a wider range of IoT and smart system projects that have no need for highly specialized features.

In conclusion, the previously mentioned alternatives are also excellent choices for smart systems, but, in my opinion, the Raspberry Pi 5 offers the best combination of cost**,** community support, and versatility.

Use cases. Let's consider examples of using the Raspberry Pi 5 in smart systems and IoT. First of all, the Raspberry Pi 5 platform has proven to be an affordable solution for home projects. Thanks to its ease of maintenance, the RP5 is used as a smart home control center: lighting, cameras, thermostats, and door locks; security surveillance system; remote home server.

In the industrial field, the RP5 can be used at the boundaries of industrial networks, where it collects sensor data (such as temperature, pressure, and vibration). It also has significant potential for agricultural applications [3]. For example, to monitor soil moisture and weather conditions, allowing farmers to optimize irrigation systems and reduce water use. A practical example of the use of Raspberry Pi devices in business is Yodeck, which uses the Raspberry Pi platform to create digital advertising signages. They developed a Python application to use with Raspberry Pi that would handle content playback, scheduling, device management, and other aspects of the service.

Self-service checkout system based on RP5. Considering the universal possibilities of the Raspberry Pi 5, we suggest a new idea: using the RP5 as the core of a self-service cash register system (fig.1). With its enhanced real-time data processing and camera support, RP5 can provide automatic product identification. We propose a system in which customers can automatically scan products as they are placed on the checkout platform. The system will process images captured by an integrated camera to identify products based on their shapes, colors, or packaging.

This new approach can make the checkout process easier. Customer experience will be better, since there will be no need for manual barcode scanning and waiting lines will be reduced. In addition, the advantages include improved customer experience. In addition, this system can reduce operating costs by allowing retailers to deploy multiple units in-store without large expenses.

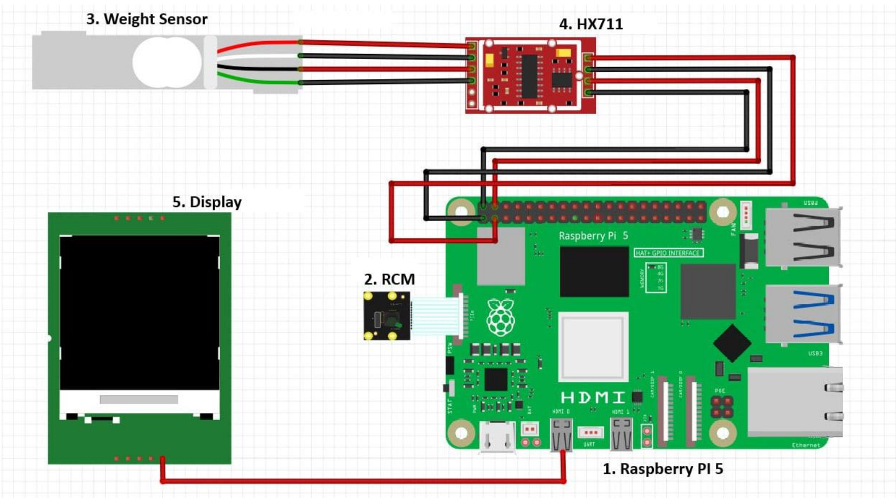


Fig. 1. Self-service checkout system based on RP5

System overview:

1. Raspberry Pi 5 serves as the central computing unit in this system, handling a variety of tasks: processing data from the weight sensor, capturing and analyzing video streams from a camera, and running algorithms to recognize items based on their shape, color, or packaging.
2. Raspberry Pi Camera Module 3 (RCM) provides the high-resolution video feed needed for product identification. The camera is capable of capturing detailed images, which are then processed by the Raspberry Pi 5 to detect and classify products.
3. The weight sensor measures the change in resistance when a mechanical load is applied to it. The weight sensor provides an additional layer of accuracy by ensuring that the product identified visually matches the expected weight of the item.
4. HX711 Module amplifies and converts analog signals to digital format that the Raspberry Pi 5 can process.
5. The display shows the customer products being scanned and their prices and allows for interaction, enabling customers to confirm their purchases.

As a recognition model, we propose to use YOLO (You Only Look Once). YOLO can process video frames and identify the product based on predefined characteristics from the training dataset.

To be more precise, YOLOv4-tiny or YOLOv5 Nano would be the best fit for this system due to their lightweight architecture, which is optimized for running on devices like the Raspberry Pi 5.

Tests. In the research, we ran several benchmarks on the Raspberry Pi 5 focusing on CPU performance and machine learning inference. It includes checks for CPU speed, file writing/reading speed, and how popular machine learning models perform. Results are shown below, in Table 1.

Table 1. Results of benchmarks and tests performed on Raspberry Pi 5

Benchmark	Raspberry Pi 5 Result	Unit of Measurement
Sysbench (Multi-core CPU)	4062	Events per second
Speedometer 2.1	67	Runs per minute
mobilenet_v2	29.7 ms.	Avg inference rate, ms.
YOLOv8s (NCNN)	101.56 ms.	Avg inference rate, ms.
Disk I/O (Read/Write Speed)	79 MB/s	Speed (MB/s)

The obtained test results show that Raspberry Pi 5 is a powerful platform for creating innovative smart systems. Performed benchmarks demonstrate a CPU event rate of 4062 per second and an average input processing time of 29.7 ms for the most popular machine learning model MobileNetV2. The results are pretty good, especially considering the size of this device.

Results. The acquired scores on performed tests prove that this device is a balanced solution between its availability, economic feasibility, and capabilities. The Raspberry Pi 5 is a reliable, cost-effective, and scalable solution to fulfill the needs of the demands of progress. On the technical side, with a size-to-power ratio and performance difference that is superior to its predecessors, the RP5 brings real-time data processing and intelligent decisions directly to the local device. Some of the tests showed a higher average performance gain of PI5 than officially announced by two to three times, where individual programs or test functions ran 10 to 18 times faster [3].

Conclusions. Based on this potential, we propose to use the Raspberry Pi 5 in self-service checkout systems. Adopting it in such systems offers several practical advantages. Firstly, its processing power and camera support enable automatic product identification using advanced algorithms. Secondly, the modularity of the RP5 allows retailers to easily upgrade and adapt the system to new devices and software updates. With the growing demand for affordable computing solutions for IoT, automation, and smart systems, this device is important due to its cost-effectiveness and versatility. Its compact design, relatively high processing power, and low power consumption make it an ideal choice for projects that require a balance between performance and cost.

The scientific novelty of this study lies in demonstrating how the Raspberry Pi 5, traditionally considered a platform for education and hobbyist projects, can be effectively used in commercial applications such as self-service checkout with real-time product recognition. The integration of visual identification through YOLO with weight sensors enhances accuracy, makes the system more reliable, reduces the possibility of product misidentification, shortens waiting times, and eliminates the need for manual barcode scanning.

1. Fezari, M., & Al Dahoud, A. (2023). Raspberry Pi 5: The new Raspberry Pi family with more computation power and AI integration (pp. 1–4). https://doi.org/10.13140/RG.2.2.13547.52009
2. Rahman, Md. W., Hossain, Md. E., Islam, R., Rashid, Md. H. A., Alam, Md. N. A., & Hasan, Md. M. (2020). Real-time and Low-cost IoT based farming using raspberry Pi. *Indonesian Journal of Electrical Engineering and Computer Science*, *17*(1), 197. https://doi.org/10.11591/ijeecs.v17.i1.pp197-204
3. Longbottom, R. (2024). Raspberry Pi 5 benchmarks and stress tests. https://doi.org/10.13140/RG.2.2.16949.24803.
4. Nasir Abdul Jalil and Koay Kian Yeik. 2020. Systems, Design and Technologies Anxieties Towards Use of Self-service Checkout. In *Proceedings of the 2019 3rd International Conference on Education and E-Learning (ICEEL '19)*. Association for Computing Machinery, New York, NY, USA, 122–127. https://doi.org/10.1145/3371647.3371664.

Serhii Hlod[1], Anastasiia Doroshenko[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, serhii.i.hlod@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, anaastasiia.v.doroshenko@lpnu.ua

METHODS AND MEANS OF REAL-TIME MILITARY OBJECTS RECOGNITION DURING MARTIAL LAW IN UKRAINE

Introduction. Object detection is one of the most important problems in computer vision. It's the basis for many other computer vision tasks, such as object segmentation, particularly in digital images, image captioning, object tracking, etc. Object detection plays a special role in real-time recognition systems. The goal of object detection is to develop computational models that provide the most fundamental information required by computer vision applications: what kind of objects they are and where they are located. Such systems not only detect and analyze movements but also provide intelligent intervention when necessary. This functionality plays a key role in a variety of scenarios, from military applications to crowd management and urban planning.

Modern trends are driving humanity to apply artificial intelligence technologies in the field of national defense and military security. These technologies are being actively implemented in modern defense systems, enabling reliable control and surveillance of military territories and borders. The most widespread application of artificial intelligence is the accurate and reliable recognition of objects or events in real time, which helps identify potential threats early and provides critical operational information for strategic decision-making and coordination of military operations. In the context of military applications, the ability to quickly and reliably recognize potentially hostile objects can be decisive [1]. Systems capable of analyzing and distinguishing between civilian and military objects significantly increase the chances of success in combat operations by minimizing civilian casualties and providing critical real-time tactical information.

Amid the ongoing hybrid large-scale war in Ukraine, the number of threats continues to rise, making it essential to rapidly modernize, enhance, and rigorously test technologies that were once deemed reliable. The safety of our citizens and the protection of critical infrastructure depend on these advancements. Machine learning (ML) technologies are becoming increasingly vital, as they can reduce the risk of human error in critical situations or assist experts in making the right decisions during crucial missions [2].

The main contributions of this paper are as follows:

- algorithms for performing tasks of recognition, detection, and early warning in military applications were analyzed;
- it has been shown that the YOLOv8 algorithm is one of the most suitable for recognition, detection, and early warning tasks in military applications.

Problem statement. ML algorithms have proven highly effective in detecting a wide range of dangerous objects in real time, particularly in military settings. For instance, they can quickly identify and track unmanned aerial vehicles (UAVs), which pose risks such as espionage or terrorist attacks on critical infrastructure. ML models can also detect missiles, small arms, armored vehicles, and even explosive devices like landmines or IEDs. This capability significantly enhances early threat detection, allowing for prompt countermeasures. Incorporating these technologies aids in managing airspace around vital facilities and identifying potential threats on the ground or sea, such as tanks, hostile vehicles, and warships. ML-based systems can analyze video feeds, radar data, and sensor inputs to ensure the safety of military installations, power grids, and other critical infrastructure.

The primary focus of this study is to perform a multi-criteria analysis of current methods and means of real-time object recognition, with a special emphasis on their military application. This analysis will help identify the optimal algorithm for efficiently detecting military threats and armored vehicles. Key performance criteria will include accuracy, detection speed, false-positive rate, and adaptability to various environmental conditions. By ensuring quick and reliable detection of dangerous objects, these ML algorithms can significantly improve response times and reinforce the overall security of essential facilities during conflicts.

Main part. Real-time object recognition consists of two essential parts: detection and tracking. Detection identifies and locates objects in individual frames, while tracking ensures the consistent following of these objects across a video feed. When using a detection model to identify objects in real time, the bounding box around an object may blink or shift between frames. This inconsistency is due to the model detecting objects frame by frame, which isn't sufficient for stable tracking – in dynamic environments. The detection phase alone does not provide the smooth, continuous object localization required for many real-world applications.

After the recognition step, the tracking phase comes into play, utilizing specialized algorithms to ensure seamless object movement tracking across multiple frames. Algorithms such as Kalman filters, SORT, or optical flow methods are typically used for this purpose. These tracking algorithms are designed to maintain consistent and accurate bounding boxes, reducing the blinking effect and providing a stable trajectory for each object, which is crucial in real-time applications like surveillance, autonomous driving, or UAV navigation.

Modern object detection algorithms can be categorized into two main types: one-stage and two-stage detectors. This classification is based on how many times the input image is processed by the network.

One-stage object detection algorithms process an input image in a single pass to predict both the presence and location of objects. This single-pass method makes them computationally efficient and well-suited for real-time applications, especially in resource-constrained environments. Although they tend to be faster, one-stage detectors like YOLO (You Only Look Once), SSD (Single Shot Detector), and RetinaNet generally offer lower accuracy compared to two-stage detectors, particularly for smaller objects. Nonetheless, their speed makes them ideal for scenarios where real-time performance is crucial, such as embedded systems, UAVs, and mobile devices.

In contrast, two-stage detectors like Faster R-CNN, which analyze images in two phases, can be slower and more resource-intensive [3]. For applications where quick response is essential and hardware limitations exist, one-stage detectors are often preferred due to their balance of speed and efficiency, making them suitable for time-sensitive and mission-critical environments [4].

It's clear that selecting the most effective algorithm is crucial for ensuring accurate and timely detection of potential threats. To evaluate different object detection algorithms, a custom dataset was compiled consisting of 2,476 images of armored vehicles captured from various perspectives (fig. 1). This diverse dataset is designed to provide a comprehensive basis for assessing the performance of the algorithms in accurately detecting and classifying armored vehicles under different conditions and viewpoints. The use of this dataset allows for a thorough analysis of each algorithm's capabilities and limitations in handling real-world scenarios involving armored vehicles.

Fig. 1. Visualization of examples from the proposed dataset

After evaluating various object detection algorithms, YOLOv8 is the most suitable option for real-time object detection in military applications. As one of the latest in the YOLO series, YOLOv8 excels with its single-stage architecture, ensuring fast and efficient processing of images and video feeds while maintaining high accuracy. This makes it particularly effective for scenarios requiring swift and reliable responses, such as battlefield surveillance and threat detection in military conflicts. Its speed, precision, versatility, and hardware adaptability make it the optimal choice for recognizing potential threats in complex and dynamic conditions.

A YOLOv8 model was specifically developed using a custom dataset of armored vehicles captured from various perspectives. This model was trained to accurately identify and classify armored vehicles, taking full advantage of YOLOv8's advanced features to optimize performance. The model achieved a solid accuracy rate of 91%, showcasing its effectiveness in precisely detecting objects in real time. Additionally, the model processed images at an impressive 45 frames per second (fps), ensuring efficient performance in time-sensitive military environments.

However, it is important to note some limitations of the YOLOv8 model. The dataset used for training, while comprehensive in terms of perspectives, lacked significant data for smaller or partially obscured vehicles. As a result, the model's ability to detect smaller or partially hidden objects is somewhat limited, impacting overall accuracy in such cases. This limitation contributed to a false-positive rate of approximately 8%, meaning that some objects were misclassified as armored vehicles. Addressing these issues through enhancements to the dataset and model training could further improve sensitivity and specificity.

Overall, despite its minor limitations, YOLOv8's 91% accuracy demonstrates its strong performance in real-time detection of armored vehicles, making it highly effective for military applications. Its balance of speed, accuracy, and adaptability proves it to be the optimal solution for recognizing and responding to threats in the challenging conditions of modern warfare.

Conclusion. Throughout the study, a comprehensive multi-criteria analysis was conducted on various object detection algorithms, including both one-stage and two-stage approaches. Based on this analysis and the performance of models developed for recognizing hazardous objects, it was concluded that the YOLOv8 algorithm is the most suitable for real-time recognition, detection, and early warning tasks in military applications. YOLOv8 stands out for its balance of speed, accuracy, and efficiency, making it particularly effective in environments where timely responses are critical.

Furthermore, the model demonstrated exceptional versatility, proving to be compatible with a wide range of hardware platforms. This adaptability is essential for deployment in mission-critical and resource-constrained environments, ensuring that the system can be seamlessly integrated into various operational settings without sacrificing performance. Overall, YOLOv8 has been identified as the optimal solution for real-time detection tasks in high-stakes military applications.

Acknowledgment. This work was realized within the framework of the Erasmus+ Jean Monnet Chair 2022 «101085612 – DataProEU – Data Protection in the EU».

1. Obelovska, K., Doroshenko, A., & Bilyk, O. (2022). Risk analysis of personal data loss in wireless sensor networks. In *CEUR Workshop Proceedings* (Vol. 3101, pp. 275–291). Proceedings of the 2nd International Workshop on Computational & Information Technologies for Risk-Informed Systems (CITRisk 2021) co-located with XXI International Conference on Information Technologies in Education and Management (ITEM 2021), Kherson, Ukraine, September 16–17, 2021.
2. Azarov, I., Gnatyuk, S., Aleksander, M., Azarov, I., & Mukasheva, A. (2023). Real-time ML Algorithms for The Detection of Dangerous Objects in Critical Infrastructures. In *CEUR Workshop Proceedings* (Vol. 3373, pp. 217–226). Proceedings of the 4th International Workshop on Intelligent Information Technologies & Systems of Information Security, Khmelnytskyi, Ukraine, March 22–24, 2023.
3. Savchuk, D., & Doroshenko, A. (2021). Investigation of Machine Learning Classification Methods Effectiveness. In *2021 IEEE 16th International Conference on Computer Sciences and Information Technologies (CSIT)*. https://doi.org/10.1109/csit52700.2021.9648582.
4. Sazonov, D. (2024, May 13). DIY for a spy: Utilizing yolov8 object detection in military operations. *Medium*. https://medium.com/ai-advances/diy-for-a-spy-utilizing-yolov8-object-detection-in-military-operations-053d787b6f62.

Serhii Holub[1], Nataliia Khymytsia[2]
[1]Cherkasy State Technological University, Cherkasy, Ukraine, s.holub@chdtu.edu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, nataliia.o.khymytsia@lpnu.ua

INFORMATION TECHNOLOGY FOR INTELLIGENT MONITORING AS A TOOL FOR HISTORICAL FORECASTING

Topicality of the study. The current challenges in the political, economic, and social spheres that the world community is experiencing cannot be solved without high-quality analytics and historical forecasting. For Ukraine, the importance of historical forecasting is paramount due to the full-scale invasion of Russia and all the geopolitical crises and transformations associated with this war. Historical forecasting is a scientific method and an interdisciplinary area of research that uses historical data and methods to predict possible scenarios for the future. Forecasting is based on a thorough analysis of historical data, the identification of patterns and cycles, and the use of various modelling and forecasting methods. The method of historical forecasting solves a number of important tasks: risk assessment (by analysing historical events, it is possible to identify potential threats and risks faced by the country); development of a development strategy (forecasting helps to formulate effective development strategies in the face of uncertainty); increasing adaptability (knowledge of historical trends allows for faster adaptation to changes in the internal and external environment); strengthening national identity (through a thorough analysis of historical events and support for national values) [1, 2].

Today, historians have a variety of tools and information technologies available to them to solve historical forecasting problems: digital databases, such as Google Ngram Viewer, which allow analysing historical texts to identify trends in the use of words, concepts, and ideas in different historical periods; machine learning, which is used to analyse large amounts of historical data and create forecasts; modelling software (e.g. STATA or R), which allows creating statistical models for forecasting based on historical data; geographic information systems (GIS) – used to visualise historical changes in geographic space, which helps to predict demographic and economic changes.

In our research, we proposed to use the methodology of modelling and expert monitoring for historical forecasting, which combines the method of extrapolation (using statistical data to predict future trends based on the past) and machine learning.

Methods and materials. The methodology of modelling and expert monitoring (MEM) is based on the combination of model synthesis methods used by the technology of multi-level intelligent monitoring [3] with methods of historical research for expert interpretation of modelling results. The MEM methodology is designed to reduce the uncertainty that arises from the subjective use of historical data to substantiate decision-making processes for the design and construction of future events. In distinction to the existing methods and tools of historical research, the MEM methodology assumes that a historian does not interpret the characteristics of historical observations, but receives for analysis the information and knowledge contained in the structure of models built by processing the results of these historical observations. Intelligent monitoring is an information technology for providing knowledge to decision-making processes by organising continuous observations and processing their results. Data mining methods are used to process the results of observations. One of the hypotheses that was put forward in the process of building the MEM methodology is the assumption that the use of expert methods of historical research to interpret information and knowledge obtained from the processing of historical data can increase the efficiency of historical forecasting [4–6].

Research results and their practical value. In this research, inductive modelling methods are used to build model synthesis algorithms (MSA) based on historical data, in particular Group Method of Data Handling (GMDH), neural networks of various topology [7], hybrid methods that combine genetic methods, neural networks and GMDH. To build a multilevel structure of the MSA of predictive models, the method of bottom-up synthesis of elements was used [8]. The task was to forecast the fertility rate of the Ukrainian population for one year ahead. The influence of the features on the modelling results was assessed. The historical data set was obtained by extracting and aggregating data from various information sources. The primary description contained seven features that were used as a set of independent variables.

$$X = \{x_1, x_2, \dots x_7\} \tag{1}$$

The elements of the set X are features that characterise the natural population growth at the previous point of observation, the economic situation of the population and Ukraine as a whole.

Figure 1 shows the accuracy characteristics of the modelling results. The accuracy of the models was assessed by the error of the forecast of natural population growth in Ukraine at the four extreme points of observation that were not used in the process of building the forecast model.

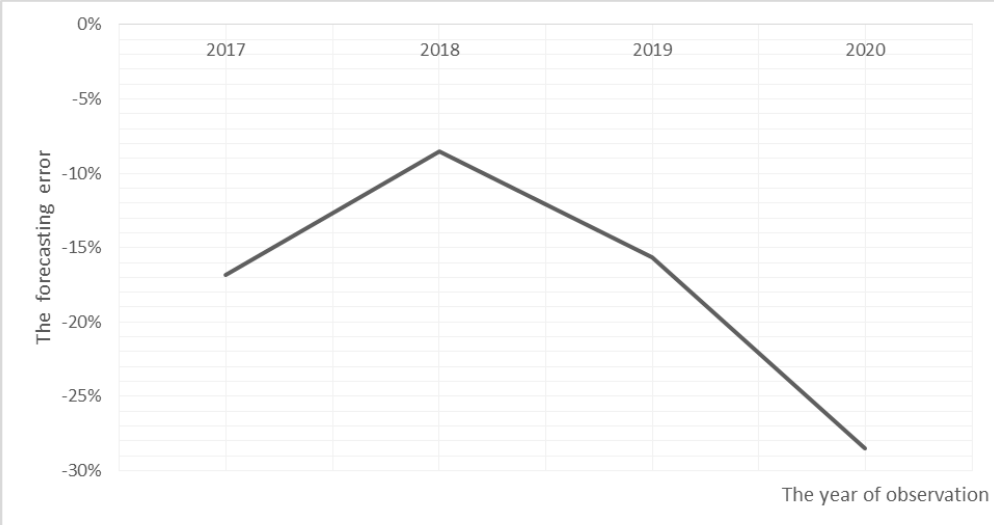


Fig. 1. Characteristics of the accuracy of modelling results

As the lead time grows, the forecasting error tends to increase.

Table 1 shows the estimated impact of the features of the input data set.

Table 1. Evaluation of the influence of the features of the input data set

№	Feature	Influence of the feature, %	The direction of change of the function with the growth of the trait
1	Natural population growth, thousand	85,95	**Growing**
2	Natural population growth, a thousand -1	0	No changes
3	Natural population growth, a thousand -2	0	No changes
4	Monitoring time	0	No changes
5	Gross domestic product*, billion UAH.	0	No changes
6	Foreign direct investment in Ukraine (end of year), USD million USD	14,05	**Growing**
7	Incomes of the population, a billion UAH	0	No changes
8	The average monthly salary, UAH.	0	No changes
9	The average monthly pension of pensioners, UAH.	0	No changes

Thus, the model's forecasting horizon is two steps. The next year's population growth increases with the growth of two indicators - the value of this indicator today and the value of foreign direct investment in Ukraine. According to the modelling results, the state creates the conditions for an increase in the birth rate through foreign investment.

Conclusions. The modelling and expert monitoring (MEM) methodology proposed in this paper provides a comprehensive combination of statistical methods and machine learning to reduce uncertainty in decision-making processes. The peculiarity of the approach is that the interpretation of modelling results is based on models built by analysing historical data, which allows for an increase in the efficiency of forecasts. In addition, the use of intelligent monitoring and Data Mining technologies makes this process more automated and objective. This approach will allow us to make more informed decisions about the future development of Ukraine, particularly in the economic, social and political spheres.

1. Sukhyi, O. (2012). Suchasni shkoly istoriiopysannia v Ukraini [Modern schools of historiography in Ukraine]. *Ukraina: Kulturna spadshchyna, natsionalna svidomist, derzhavnist: Zbirnyk naukovykh prats, 21*, 710-723. Lviv: Instytut ukrainoznavstva im. I. Krypiakevycha NAN Ukrainy.
2. Ustiyanovich, T., & Khymytsia, N. (2018). Application of machine learning algorithms for the study of Galicia population’s social characteristics in the interwar period (1919-1939). In *8th International Academic Conference of Young Scientists “Humanities and Social Sciences 2018” (HSS-2018)* (pp. 96-101). Lviv: Lviv Polytechnic Publishing House.
3. Holub, S. V., & Khymytsia, N. O. (2012). Vidobrazhennia konsolidovanoi informatsii ekonomichnykh pokaznykiv rehionu u strukturi bahatorivnevykh modelei [Representation of consolidated information of regional economic indicators in the structure of multilevel models]. *Visnyk Skhidnoukrainskoho natsionalnoho universytetu imeni Volodymyra Dalia, 8*(179), Part 1, 122-128.
4. Holub, S., & Khymytsia, N. (2016). The use of multi-level modeling in the cliometric studies process. In *Modern Problems of Radio Engineering, Telecommunications and Computer Science: Proceedings of the 13th International Conference on TCSET 2016* (pp. 733–735). https://doi.org/10.1109/TCSET.2016.7452166
5. Golub, S., & Khymytsia, N. (2018). The method of cliodinamik monitoring. In *Proceedings of the 2018 IEEE 2nd International Conference on Data Stream Mining and Processing (DSMP 2018)* (pp. 223–226). https://doi.org/10.1109/DSMP.2018.8478459
6. Khymytsia, N., & Holub, S. (2019). Intellectual analysis of the results of the cliometric monitoring. *Matematychni mashyny i systemy [Mathematical Machines and Systems], (4)*, 87–92. Retrieved from http://immsp.kiev.ua/publications/articles/2019/2019_4/04_Khimitsa_19.pdf
7. Nemchenko, V. V., & Nechyporenko, V. Yu. (2010). Koordynatsiia struktury bahatorivnevykh monitorynhovykh system za dopomohoiu neiromerezh [Coordination of the structure of multilevel monitoring systems using neural networks]. In *Pyata naukovo-praktychna konferentsiia z mizhnarodnoiu uchastiu "Matemytchne ta imitatsiine modeliuvannia system. MODS '2010". Tezy dopovidei* (pp. 205-206). Kyiv.
8. Kharchenko, O. V., Holub, S. V., & Zhyriakova, I. A. (2016). Udoskonalennia metodu vyskhidnoho syntezu elementiv v informatsiinii tekhnolohii bahatorivnevoho monitorynhu mobilnoho robota [Improvement of the bottom-up synthesis method in the information technology of multilevel monitoring of a mobile robot]. *Matematychni mashyny i systemy [Mathematical Machines and Systems], (3)*, 41-47.

Artem Kazarian[1], Vasyl Teslyuk[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, artem.h.kazarian@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, vasyl.m.teslyuk@lpnu.ua

IMPLEMENTATION OF AI-DRIVEN INFORMATION SYSTEM FOR SCIENTIFIC AND BUSINESS PROJECT IDEAS GENERATION

Introduction. Various systems help users generate ideas based on data mining [1, 2], recommendation systems, and personalized suggestion algorithms [3, 4]. However, few incorporate AI-driven LLMs to develop project ideas specifically tailored to user experience and preferences. Some related work includes recommendation engines for business ventures, creativity-enhancing software, and systems that match people with scientific problems or business opportunities.

Materials and methods. The core of the proposed system lies in its ability to transform user input into an effective prompt for the large language model. Users input data via a web interface, answering questions categorized into interests, skills, experience, and preferences. This data is structured into a prompt that queries the LLM for relevant project ideas. The backend architecture integrates this process with AI models, which handle natural language processing and idea generation. The system employs prompt engineering techniques to maximize the relevance and creativity of generated content.

The system is implemented as a web-based platform allowing users to interact via a simple and intuitive interface. Data collection is done through a structured questionnaire. The AI model, integrated via an API, receives the user's responses and generates a customized project idea. The idea is then presented to the user as a project name, description, and list of features.

Results. The system generates results in the form of detailed project ideas, each designed to suit the specific interests and skills of the user. These ideas include a project name, a brief description, key functions, and potential areas for further development. Several case studies will be presented to illustrate the system's capabilities, demonstrating how it can adapt to various user profiles. The output data will be discussed in terms of accuracy, relevance, and usability for real-world application in both business and scientific settings. The screen with generated results is provided in Figure 1.

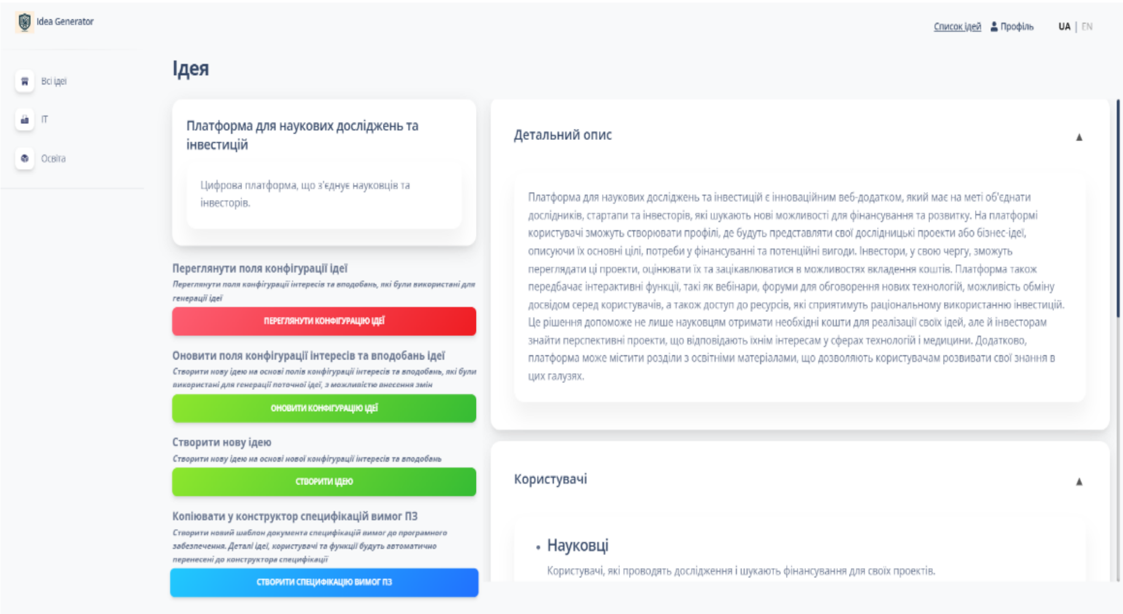


Fig. 1. Visual user interface for presentation of generated ideas

Using controls of the user interface, users can save generated ideas in their profiles (drafts will be removed from the database in 100 days after generation automatically), review the questionnaire form with answers that were used for idea generation, can edit answers in previously filled questionnaire form for generation of new idea.

The system consists of the following components: front-end, API, server-side, database, and the large model service. The front-end is implemented using the Vue.js JavaScript framework and is hosted on an AWS S3 bucket. The API is based on AWS API Gateway and routes front-end requests to the server-side. The server-side is responsible for handling requests from the front-end, managing the logic for creating prompts for the large language model, performing database read and write operations, and communicating with the large language model service [5, 6]. A brief structure of the system architecture is presented in Figure 2.

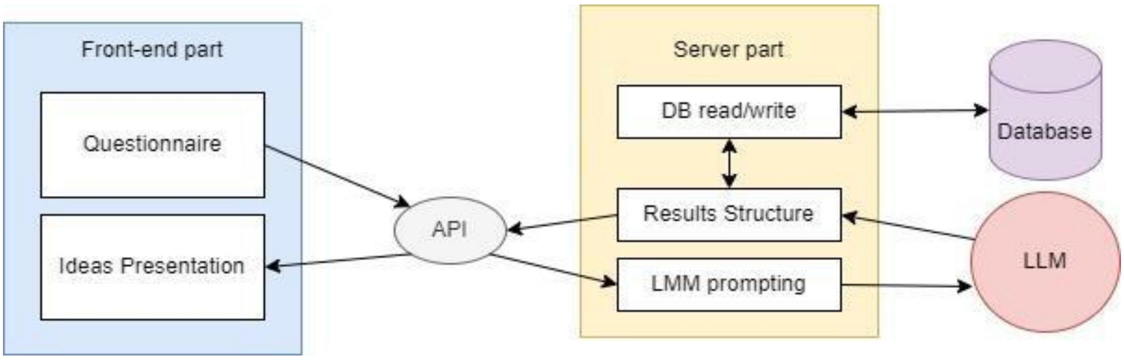


Fig. 2. Structure of the system architecture

Conclusions. This paper has introduced a novel system for automated idea generation that integrates artificial intelligence, specifically large language models, to produce tailored project concepts for users in scientific and business domains. The system facilitates the early stages of innovation and research by aligning project ideas with a user's interests and experience. Future improvements may include refining prompt-generation techniques, expanding the user interface, and enhancing idea diversity.

Acknowledgement. This work was realized within the framework of the program Erasmus+ Jean Monnet Module «Augmented Reality for Education: implementation of European experience» (101085772 – AR4EDU – ERASMUS-JMO-2022-HEI-TCH-RSCH).

1. Girotra, K., Meincke, L., Terwiesch, C., & Ulrich, K. (2023). Ideas are Dimes a Dozen: Large Language Models for Idea Generation in Innovation. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.4526071
2. Si, C., Yang, D., & Hashimoto, T. (2024). Can LLMs Generate Novel Research Ideas? A Large-Scale Human Study with 100+ NLP Researchers. https://doi.org/10.48550/arXiv.2409.04109
3. Jeong, C. (2024). A Study on the Implementation Method of an Agent-Based Advanced RAG System Using Graph. https://doi.org/10.48550/arXiv.2407.19994
4. Sánchez-Díaz, X., Ayala-Bastidas, G., Fonseca, P., & Garrido, L. (2018). A knowledge-based methodology for building a conversational chatbot as an intelligent tutor: 17th Mexican International Conference on Artificial Intelligence, MICAI 2018, Guadalajara, Mexico, October 22–27, 2018, Proceedings, Part II. *Springer*. https://doi.org/10.1007/978-3-030-04497-8_14
5. Lyu, W., Wang, Y., Chung, R., Sun, Y., Zhang, Y., & Zhang, Y.-U. (2024). Evaluating the effectiveness of LLMs in introductory computer science education: A semester-long field study. *ACM Transactions on Computing Education*, 24(4), Article 11. https://doi.org/10.1145/3657604.3662036
6. Jeong, C. (2023). A Study on the Implementation of Generative AI Services Using an Enterprise Data-Based LLM Application Architecture. *Advances in Artificial Intelligence and Machine Learning*, 3, 1588-1618. https://doi.org/10.54364/AAIML.2023.1191

Dmytro Khomutnyk[1], Andrii Pukach[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, d.khomutnyk@gmail.com
[2]Lviv Polytechnic National University, Lviv, Ukraine, andriy.i.pukach@lpnu.ua

AN OPTIMIZED METHOD DEVELOPMENT FOR DESCRIBING CLOUD INFRASTRUCTURES

Problem statement. More and more organizations are utilizing cloud providers to host their internally developed multi-component distributed computing systems and perform distributed data calculations [1]. As cloud technologies become more popular, the need for convenient and fast interaction with these technologies is also growing. However, there is a certain entry barrier when it comes to using cloud resources and building cloud infrastructures, which is primarily caused by the complexities of deploying and configuring cloud resources as well as their network communication.

In the past, certain cloud automation tools have already been developed. They help to automate the deployment and configuration of resources on cloud infrastructures. Cloud providers themselves offer tools for declarative descriptions of cloud infrastructures in the form of source code. These include tools like AWS CloudFormation for Amazon Web Services [2], Azure Resource Manager for Microsoft Azure [3], and Cloud Deployment Manager for Google Cloud Provider [4].

While these tools allow us to describe the desired cloud infrastructure and the automatic deployment of corresponding resources, they are all tied to specific cloud providers. Their main drawbacks are the low-level abstractions provided for describing cloud resources and the large number of details required to describe individual resources. Such detailed descriptions complicate and slow down the process of defining cloud infrastructures for distributed computing systems and lead to many small errors in the whole process. The absence of standard methods for describing cloud infrastructures among different cloud providers is explained in the paper [5].

Therefore, more generalized tools for describing cloud infrastructure have begun to emerge on the market. The most popular tools are Terraform [6] and Ansible [7]. These tools remove the need to learn different cloud automation tools from different cloud providers. Users are able to learn a single description language and use it to create cloud infrastructures on different cloud providers. These tools also offer the capability to create multi-provider cloud infrastructures to deploy the resources necessary for a single computing system.

However, the aforementioned cloud automation tools do not solve the issue of the complexity of cloud infrastructures and individual resources from different cloud providers. To use Terraform at even a basic level, users need to have a profound understanding of cloud resource creation, configuration, and intercommunication. Additionally, users must be aware of and take into account the configuration intricacies of similar resources (such as virtual machines) from different cloud providers.

A survey on cloud service description [8] investigates possible ways and methods to unify the cloud description methods and tools among different cloud providers. It defines a foundation for a standardized method of describing cloud infrastructures that is agnostic to specific cloud providers.

Proposed solution. The author has proposed a method for describing cloud infrastructure that addresses the problem of excessive complexity in the description of basic infrastructures. This method hides the intricacies of fine-tuning individual resources behind higher-level abstractions compared to the existing description methods. It also allows users to omit defining connections between different resources explicitly, as these will be generated automatically. The method is described in detail in the author's publication [9]. It contains a detailed explanation of all available statements and abstractions used in the method and how they correspond to the actual cloud infrastructure.

For illustrative comparison, Figure 1 shows the description of identical infrastructures for the cloud provider AWS using the following methods (from left to right): CloudFormation for Amazon WebServices, Terraform, and the proposed method.

```yaml
AWSTemplateFormatVersion: 2010-09-09
Resources:
  ExampleVPC:   (1)
    Type: AWS::EC2::VPC
    Properties:
      CidrBlock: 172.22.0.0/16 (2)
  ExampleSubnet:  (3)
    Type: AWS::EC2::Subnet
    Properties:
      VpcId: !Ref ExampleVPC (4)
      CidrBlock: 172.22.10.0/24 (5)
      AvailabilityZone: eu-north-1a (6)
  ExampleSecurityGroup: (7)
    Type: AWS::EC2::SecurityGroup
    Properties:
      VpcId: !Ref ExampleVPC (8)
      GroupName: example-security-group (9)
      GroupDescription: 'Allow all ingress traffic'
      SecurityGroupIngress:
        - IpProtocol: -1   (10)
          FromPort: -1
          ToPort: -1
          CidrIp: 0.0.0.0/0

  ExampleNetworkInterface:  (11)
    Type: AWS::EC2::NetworkInterface
    Properties:
      SubnetId: !Ref ExampleSubnet (12)
      GroupSet:
        - !Ref ExampleSecurityGroup (13)
      PrivateIpAddresses:
        - PrivateIpAddress: 172.22.10.4 (14)
          Primary: 'true'
  ExampleEC2: (15)
    Type: AWS::EC2::Instance
    Properties:
      ImageId: 'ami-078e13ebe3b027f1c'(16)
      InstanceType: t3.micro (17)
      NetworkInterfaces:
        - NetworkInterfaceId: !Ref ExampleNetworkInterface (18)
          DeviceIndex: '0'
```

```hcl
resource "aws_vpc" "ExampleVPC" { (1)
  cidr_block = "172.22.0.0/16" (2)
}

resource "aws_subnet" "ExampleSubnet" { (3)
  vpc_id            = aws_vpc.ExampleVPC.id (4)
  cidr_block        = "172.22.10.0/24" (5)
  availability_zone = "eu-north-1a" (6)
}

resource "aws_security_group" "ExampleSecurityGroup" { (7)
  name        = "example-security-group" (9)
  description = "Allow all ingress traffic"
  vpc_id      = aws_vpc.ExampleVPC.id (8)

  ingress {
    protocol    = "-1"
    from_port   = 0      (10)
    to_port     = 0
    cidr_blocks = ["0.0.0.0/0"]
  }
}

resource "aws_network_interface" "ExampleNetworkInterface" { (11)
  subnet_id       = aws_subnet.ExampleSubnet.id (12)
  private_ips     = ["172.22.10.4"] (14)
  security_groups = [aws_security_group.ExampleSecurityGroup.id] (13)
}

resource "aws_instance" "ExampleEC2" { (15)
  ami           = "ami-078e13ebe3b027f1c" (16)
  instance_type = "t3.micro" (17)

  network_interface {                                                 (18)
    network_interface_id = aws_network_interface.ExampleNetworkInterface.id
    device_index         = 0
  }
}
```

```
VPC ExampleVPC {(1)
  cidr_block = "172.22.0.0/16" (2)
  region = "eu-north-1"(6)

  Subnet ExampleSubnet { (3)

    EC2 ExampleEC2 { (15)
  (16) ami = "ami-078e13ebe3b027f1c"
  (17) instanceType = "t3.micro"

  (11) networking {
        ingress = all (10)
      }
    }
  }
}
```

Fig. 1. Comparison of cloud infrastructure description methods.
Left to right: CloudFormation, Terraform, and the proposed method.

In this example, the most basic cloud infrastructure necessary for running user code on a remote virtual machine is described. A private cloud network (1) with a designated IP address block (2) is declared.

In the private cloud, a subnet (3) is defined with the explicit designation to the cloud (4) and an assigned block of IP addresses (5). The physical region where the subnet and all corresponding network interfaces are located is also specified (6).

To configure access to the subnet's resources, a set of security rules (7) is defined, including its association with the private cloud (8), the name of the security group (9), and the specific rule itself, which grants inbound access from all IP addresses and network ports (10). Next, a network interface (11) is declared, connected to the previously defined subnet (12) and security rules (13). This network interface is assigned a specific IP address within the block of IP addresses designated for the subnet (14).

Finally, the server itself — a virtual machine (15) — is defined with the specified operating system image (16), server machine type (17), and network interface (18).

This is the most basic cloud infrastructure configuration that allows one to deploy and run a computation system on cloud resources. To define such infrastructure in Terraform, for example, the user needs to know many implementation details: define subnets with designated IP addresses, correctly define security rules to allow specific incoming and outgoing traffic, explicitly define network interfaces and attach them to virtual machines, find out correct image names of desired operational systems, and so on. It takes time to learn all such details when the main priority of the user is to deploy their developed system to the Internet simply.

As shown in Figure 1, the proposed method allows for the description of such basic cloud infrastructures with significantly fewer configuration details for resources and their interconnections. The description using CloudFormation takes 41 lines of code and includes numerous details about the configuration of individual resources. The Terraform description takes 38 lines of code and presents all resources in a similar generalized manner, simplifying the readability of the description and the use of resources from different cloud providers.

The description using the proposed method requires only 16 lines of code and contains a minimal amount of necessary details to describe a complete cloud infrastructure. The user can concisely describe the desired cloud infrastructure in general terms, while all configuration details for individual resources and their interconnections will be generated automatically.

Conclusions. The scientific novelty of the proposed method lies in the improvement of the presented existing methods for describing cloud infrastructures by a significant reduction of code required to describe cloud infrastructures by using higher-level abstractions to represent cloud resources.

The proposed method will not eliminate the need for organizations to hire highly qualified employees for specific and optimal configurations of cloud infrastructures for developed computation systems. However, it will significantly speed up the initial development and deployment of distributed systems for purposes such as testing or initial system launches. This simplification can be crucial for students studying distributed systems, beginner developers, or organizations just starting and lacking the resources to afford highly qualified cloud infrastructure specialists.

It was shown that the proposed method allows for cloud infrastructure descriptions with at least half the number of lines compared to existing methods and languages for describing such infrastructures, which improves the existing methods. As the number of resources and their interconnections increases, the difference in the number of lines of code will grow even larger. For the presented example of basic cloud infrastructure, the definition using CloudFormation takes 41 lines of code, the Terraform description takes 38 lines of code, and the description with the proposed method takes just 16 lines. That makes the proposed method more optimized in terms of code size and requirements of a profound understanding of specific cloud providers and their resources.

It is possible to tune the generated cloud infrastructure to the specific needs of users. Some users don't require or can't afford to have load balancing, autoscaling, and complex interconnections between the virtual machines and deployed services on them. The high-level abstractions used in the proposed description method potentially enable the use of various optimization techniques for cloud infrastructures, aimed at cost-saving when utilizing cloud provider services or accelerating interactions between different cloud resources (and thus speeding up the computation system as a whole).

With the increasing popularity of generative AI, more and more tools provide human-like interactions as their input interface to do some specific tasks. So there is also a potential to utilize generative neural networks to generate cloud infrastructure descriptions with a proposed method based on natural language requests from users.

1. Ker, D. (2021). Measuring cloud services use by businesses. *OECD Digital Economy Papers, No. 304*. OECD Publishing, Paris. https://doi.org/10.1787/71a0eb69-en.
2. Amazon Web Services. (n.d.). *AWS CloudFormation: User guide*. Amazon Web Services, Inc. https://docs.aws.amazon.com/cloudformation/
3. Microsoft. (n.d.). *Azure Resource Manager*. Microsoft. https://azure.microsoft.com/en-us/get-started/azure-portal/resource-manager
4. Google Cloud. (n.d.). *Deployment Manager documentation*. Google. https://cloud.google.com/deployment-manager/docs
5. Monteiro, A., Pinto, J. S., Teixeira, C., & Batista, T. (2011). Cloud interchangeability Redefining expectations. In *Proceedings of the 1st International Conference on Cloud Computing and Services Science CLOSER* (Vol. 1, pp. 180-183). Noordwijkerhout, Netherlands. https://doi.org/10.5220/0003393301800183
6. HashiCorp. (n.d.). *Terraform documentation*. HashiCorp. https://developer.hashicorp.com/terraform/docs
7. Red Hat. (n.d.). *Ansible documentation*. Red Hat. https://docs.ansible.com/
8. Ghazouani, S., & Slimani, Y. (2017). A survey on cloud service description. *Journal of Network and Computer Applications*, 91, 61–74. https://doi.org/10.1016/j.jnca.2017.02.009
9. Khomutnyk, D., & MarchenkoO. (2022). High-level technique for description of cloud infrastructure resources. *Computer-Integrated Technologies: Education, Science, Production*, (48), 117-123. https://doi.org/10.36910/6775-2524-0560-2022-48-18

Oleh Kopylchak[1], Iryna Kazymyra[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, oleh.a.kopylchak@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, iryna.y.kazymyra@lpnu.ua

HYBRID ALGORITHM DEVELOPMENT FOR EFFICIENT TRAINING COURSE RECOMMENDATIONS

Introduction. In today's world, where the amount of information is growing rapidly and the role of education is constantly increasing, it is becoming important to implement automated systems to personalize learning processes. Traditional recommendation methods are not always able to provide the necessary adaptability, especially when user data is limited. This creates a need to improve recommender systems, which will improve the quality of learning services. By using hybrid algorithms that combine collaborative filtering and neural networks, it is possible to increase the efficiency of such systems, while maintaining accuracy and adaptability even in the face of insufficient data. This research focuses on the development of a hybrid algorithm for course recommendation, which allows for a more personalized and effective educational experience.

Problem statement. The object of research is algorithms and models for generating personalized training course recommendations. This research focuses on the development of a model and an algorithm to enhance the adaptability and effectiveness of recommendations in educational settings. The subject of study is the integration of a hybrid algorithm that combines the strengths of collaborative filtering and neural networks to address the limitations of existing recommendation methods, particularly under data constraints.

The purpose of the research is to develop a hybrid recommendation algorithm that provides high accuracy and flexibility across various data volumes. The goal is to improve the personalization of learning recommendations, ensuring that the algorithm remains effective even with limited user data. This research aims to enhance the efficiency of educational recommendation systems by introducing a model that adapts to the amount and quality of available data, optimizing the learning experience for users in diverse educational environments.

The dataset [1] used in this study consists of approximately 23,000 courses in various categories, including business, design, music, and web development. The dataset contains information about 1,000 users, including their ratings and engagement with the courses. This allows for a more in-depth analysis of user behavior and preferences, which is critical for developing a personalized recommendation algorithm. This dataset, rich in both course data and user engagement data, is the basis for training and evaluating the hybrid recommender algorithm. The inclusion of user ratings and attendance allows for a more personalized recommendation system, ensuring that the algorithm can handle different amounts of data while maintaining a high level of accuracy and relevance for different categories and types of user behavior.

Main part. In the scope of this study, a hybrid algorithm that combines collaborative filtering and neural networks was developed to improve the accuracy of course recommendations. The algorithm provides high efficiency both with a large amount of data and with a limited amount of information about users. The main feature of this approach is the automatic switching between methods depending on the available data, which allows optimizing the recommendation process.

At the initial stage of the work, the data was filtered and normalized to ensure high-quality training and test samples. Categorical data, such as course difficulty and subject matter, were converted into numerical data using a coding method to allow for further modeling. After that, a detailed analysis of the data was carried out, including the distribution of key characteristics such as the number of reviews, course rating, number of lectures, price, and difficulty level. A correlation matrix was also built to study the relationships between features, which helped to identify key dependencies between different course characteristics.

The deep neural network was developed as one of the model components. It works on the basis of user interaction with training courses. The network processes multidimensional data, such as course difficulty, ratings, and user behavior, enabling more accurate recommendations by learning hidden patterns in user interactions [2]. The neural network has the following architecture:

- Input layer: 25 neurons representing aggregated data on user interaction with courses.
- Hidden layers: three layers of 50 neurons each, where the ReLU activation function is used.
- Output layer: 1 neuron responsible for predicting the course rating for the user.

The RMSprop optimizer and the MSE (mean squared error) loss function were used. Below are the final results of the model's performance.

Table 1. Metrics

Metric	Value
Mean Squared Error (MSE)	0.056
Mean Absolute Error (MAE)	0.1719
Training time	~8300 sec
Mean percentage error	17%

Conclusions. In this research, a hybrid recommendation algorithm combining collaborative filtering and neural networks was developed to enhance the adaptability and accuracy of educational course recommendations. The key advantage of the proposed approach lies in its ability to automatically switch between collaborative filtering and neural networks based on the available amount of data. This allows the algorithm to maintain high accuracy even with minimal user data, meaning the system can generate relevant recommendations with very limited input from new users (fewer than 5 data entries), effectively solving the cold start problem, while maximizing efficiency with larger datasets.

The dataset used in the study, consisting of approximately 23,000 courses and data from 1,000 users, enabled a comprehensive analysis of user behavior and course characteristics. Through the filtering, normalization, and transformation of categorical data into numerical formats, a deep neural network was successfully integrated into the hybrid model. The network was trained using the RMSprop optimizer and MSE loss function, ensuring the minimization of errors between predicted and actual ratings.

The final results demonstrated the hybrid algorithm's ability to achieve over 85% accuracy in course recommendations, even with limited user interaction data. This approach outperforms traditional methods, such as using collaborative filtering [3] and a content-based [4] method individually, both of which showed lower accuracy when applied separately. By combining these methods in a hybrid model, the algorithm achieves superior results, offering greater flexibility and adaptability, ultimately making it a more effective tool for enhancing user engagement and learning outcomes.

Overall, this study has shown that hybrid algorithms are a promising solution for addressing the limitations of traditional recommendation systems, particularly in educational environments. The combination of collaborative filtering and neural networks offers a balanced approach that can handle different volumes of data while maintaining high precision in personalized recommendations. Future work may focus on refining the model further, incorporating additional features to enhance its robustness and exploring its application in other domains beyond education.

1. Udemy Courses Analysis Dataset. (n.d.). *Kaggle: Your Machine Learning and Data Science Community*. https://www.kaggle.com/code/sreelathar/udemy-courses-analysis
2. Pérez-López, D., Dueñas-Lerín, J., Ortega, F., & González-Prieto, Á. (2023). New trends in artificial intelligence for recommender systems and collaborative filtering. *Applied Sciences, 13*(15), 8845. https://doi.org/10.3390/app13158845
3. Increasing the Effectiveness of Prediction in Recommendation Engines Based on Collaborative Filtering. (2024). *Bilad Alrafidain Journal for Engineering Science and Technology, 3*(1), 47-58, doi: http://dx.doi.org/10.56990/bajest/2024.030104
4. Dacrema, M. F., Cremonesi, P., & Jannach, D. (2019). Are we really making much progress? A worrying analysis of recent neural recommendation approaches. *Proceedings of the 13th ACM Conference on Recommender Systems*. https://doi.org/10.1145/3298689.3347058.

Taras Korol[1], Anna Gurynovych[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, taras.korol.pp.2023@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, anna.s.hurynovych@lpnu.ua

TRAFFIC MANAGEMENT AND LIGHT CONTROL SYSTEM FOR SOLVING CONGESTION PROBLEM IN A SMART CITY

Introduction. Traffic congestion has become one of the most critical challenges in urban environments, particularly when cities continue to expand and populations grow. In the context of smart cities, the management of transportation systems is essential to ensure the smooth movement of people and goods. One of the traffic management main components is the effective use of traffic lights.

However, the traditional traffic light systems, which operate on pre-set time sequences, have proven to be insufficient in the face of the rising traffic volumes and the need for more sustainable solutions. This work explores the implementation of intelligent traffic management systems, focusing on traffic light control to alleviate congestion and enhance the efficiency of mobility in smart cities.

Description. Multiple aspects of our lives have been made more efficient by the use of technology, but the transport industry has not recognized the potential for a long time. These advancements are now inevitable as the road environment is becoming increasingly more difficult to navigate due to its growing complexity. This context has prompted the IT sector to become interested in smart cities. A smart city is a city that uses information and communication technologies to improve the quality of city services or reduce costs.

Traffic lights are already in place to control the traffic flow, but they are becoming really inefficient, limited by their design. The pre-set time change between the green, orange, and red lights operates regardless of the changing conditions, which leads to longer waiting, additional fuel consumption, and air pollution. To better meet road management needs, it is necessary to create more efficient, economical, and sustainable transport systems through the improvement of traffic signals.

Subtasks. To tackle these challenges and optimize the traffic flow in urban environments, prioritizing the implementation of intelligent traffic management systems is essential. Specifically, traffic signals need to evolve into adaptive systems that respond dynamically to real-time traffic conditions. Achieving this goal involves several crucial steps. First, traffic volume should be calculated using advanced classifiers capable of accurately analyzing vehicle density and flow patterns. Second, an algorithmic control system needs to be developed to overcome traffic jams by adjusting signal timings based on real-time data. These two subtasks are crucial for creating a more efficient, sustainable, and intelligent transport system that can alleviate urban road congestion.

Results. The intelligent traffic light system detects the presence, assesses traffic levels, and responds accordingly. Intelligent traffic systems are designed to prevent congestion at traffic lights when there are free roads.

The agent-based traffic management system uses multiple agents embedded in intersection controllers to communicate and dynamically adjust signal timings based on real-time traffic conditions. These agents interact with each other and surrounding intersections, responding to traffic flows to create a more adaptive system. By responding to changing conditions, the system reduces waiting times, fuel consumption, and overall congestion [1].

A key study supporting this approach is the Adaptive Traffic Control System (ATCS). ATCS is a traffic management solution that adjusts traffic light timings based on real-time traffic demand. This system consists of both hardware and software components, creating an adaptive solution to fluctuating traffic patterns [3]. The approach used in ATCS splits the problem into two parts: first, the traffic volume is estimated using classifiers under the Viola-Jones Object Detection System, leveraging OpenCV libraries for real-time traffic analysis. OpenCV was chosen for its speed and accuracy in processing large datasets, making it ideal for time-sensitive traffic management. This volume estimation provides the necessary data to adjust the traffic signals efficiently.

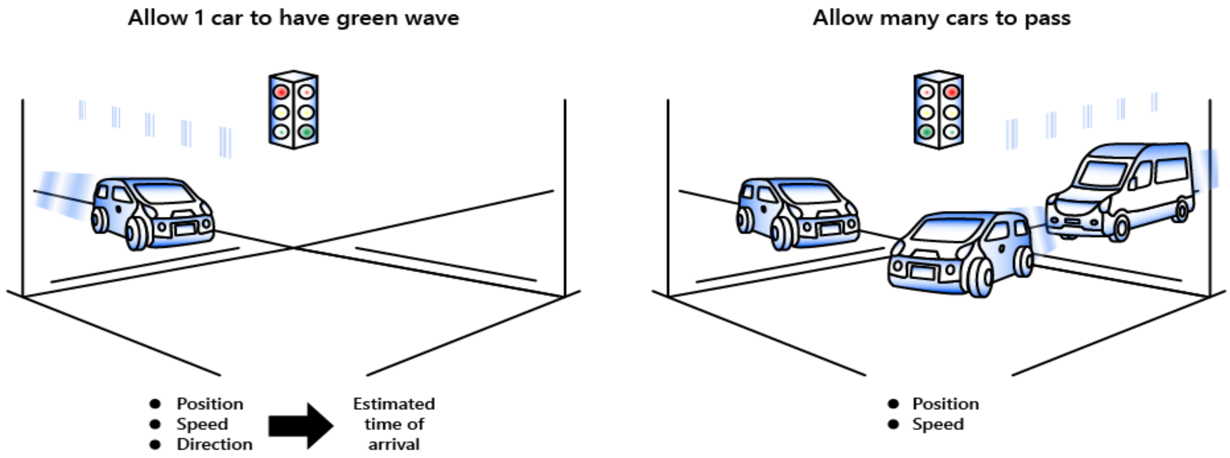


Fig. 1. Comparison of the smart and non-smart traffic light efficiency [2]

The second part of the ATCS approach focuses on controlling traffic at intersections through a control algorithm, designed to reduce traffic jams. The system includes features specifically aimed at emergency vehicles, shortening signal cycle times and ensuring their swift movement through busy roads. This adaptive approach enables a more dynamic and responsive system, reducing congestion and improving the overall traffic flow, particularly in high-density urban environments.

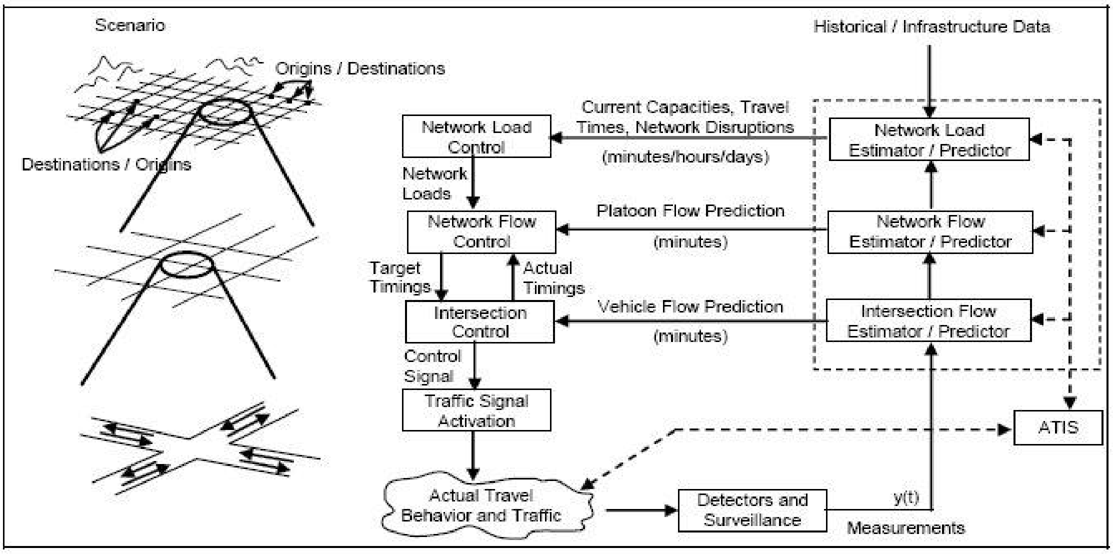


Fig. 2. The RHODES Hierarchical Architecture [4]

Another significant advancement in intelligent traffic systems is the Adaptive Traffic Light Control System (ATLCS). ATLCS seeks to synchronize traffic lights across adjacent intersections, with the primary goal of reducing the "stop and go" phenomenon common in traditional traffic systems. By dynamically delaying the moments when each traffic light turns green, based on the number of vehicles waiting at each junction, ATLCS creates a coordinated flow of traffic. This synchronization allows vehicles to travel greater distances without needing to stop frequently, reducing both travel time and fuel consumption.

The performance of the ATLCS has shown impressive results. According to its evaluation, vehicles traveling in synchronized directions experienced up to a 39% reduction in travel time compared to traditional fixed-time systems. Additionally, there was a 17% improvement in overall traffic flow across the entire road network. These findings highlight the effectiveness of adaptive systems in managing traffic in urban environments and reducing congestion, which is typical for many modern cities.

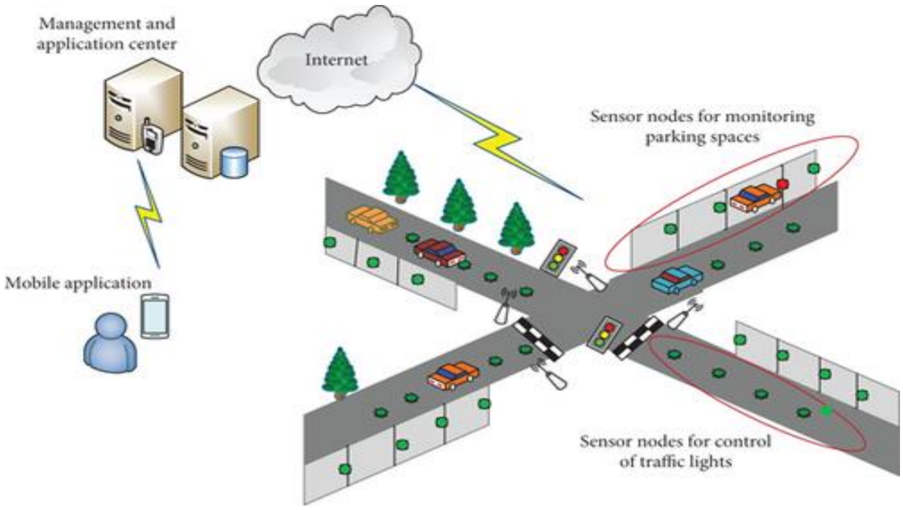


Fig. 3. Example of intelligent traffic control system [5]

Conclusions. In conclusion, this paper highlights the critical role of intelligent traffic management systems, such as the agent-based traffic signal timing system, the Adaptive Traffic Control System (ATCS), and the Adaptive Traffic Light Control System (ATLCS), in addressing the growing problem of urban congestion. By leveraging real-time data and adaptive algorithms, these systems offer a more efficient, economical, and sustainable approach to traffic control. They significantly reduce travel times, fuel consumption, and emissions, while improving road safety and emergency vehicle response times. As cities continue to expand and embrace smart technologies, the adoption of these systems presents a promising opportunity to transform urban mobility and enhance the quality of life in increasingly crowded environments.

1. Wided, A., Assia, B., & Fatima, B. (2023, July). Traffic management system and traffic light control in smart city to reduce traffic congestion. *International Journal of Automation and Smart Technology*, *13*(1).
2. AutoPi.io. (2021, September). Smart traffic light control system – intelligent mobility. https://www.autopi.io/blog/smart-traffic-light-management-shaping-smart-city/
3. Shankaran, R. S., & Rajendran, L. (2021, January). Real-time adaptive traffic control system for smart cities. In *2021 International Conference on Computer Communication and Informatics (ICCCI).* IEEE Xplore. https://ieeexplore.ieee.org/document/9402597
4. Federal Highway Administration. (2005, October). Traffic control systems handbook: chapter 3 page 2. *U.S. Department of Transportation*.
5. Hilmani, A., Maizate, A., & Hassouni, L. (2020, September). Automated real-time intelligent traffic control system for smart cities using wireless sensor networks. *Wireless Communications and Mobile Computing*, *2020* (1).

Roman Lysenko[1], Oleksa Skorokhoda[2]
[1]EPAM, Lviv, Ukraine, rlysenko@gmail.com
[2]Lviv Polytechnic National University, Lviv, Ukraine, oleksa.skorokhoda@gmail.com

INTELLIGENT MEANS OF DATA COLLECTION

Relevance of the research. Traditional methods of data collection, such as manual surveys and observational techniques, have been widely used across various industries. Intelligent data collection refers to the use of advanced technologies, such as artificial intelligence (AI), machine learning (ML), Internet of Things (IoT), and big data analytics, to automate and enhance the process of gathering, processing, and analyzing data. Artificial intelligence (AI) is not just an attempt to understand, but also to build intelligent entities [1]. Main approaches for building AI concepts were identified as intelligent agents, which represent the original base for building different kinds of computation intelligence. An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators [1].

Each such agent implements a function that maps percept sequences to actions, and we cover different ways to represent these functions [2].

Essentially, sensors and actuators are main conceptual features that allow AI to contact with a certain environment. Each agent processes and stores data according to its own needs. The integration of AI and machine learning allows for predictive insights and the identification of patterns that would be difficult to detect manually. Moreover, intelligent systems can adapt and improve over time as they learn from the data they collect, offering continuous improvements in accuracy and efficiency.

A general problem related to the development of agents is a universal vision of sensors and actuators concept. In practice, when the environment is clearly defined, it is possible to introduce specific implementations of sensors that collect needed data about the environment. Building specific sensors for data collection requires knowledge about environment structure, behavior, and life cycle. In a nutshell, it requires the specific creation of sensors and their integrations with agents.

One of the key drivers behind the adoption of intelligent means of data collection is the exponential growth in data generated by modern digital platforms and devices. However, the shift to intelligent data collection raises concerns regarding data privacy, security, and ethical implications. As intelligent systems become more pervasive, ensuring that data is collected, stored, and used responsibly is paramount.

Thus, the primary problem addressed by this research is the need for scalable, accurate, and secure intelligent data collection systems capable of handling diverse and large datasets while mitigating challenges related to data quality, privacy, and technical limitations.

The methods and materials. Intelligent agents are autonomous software programs designed to perform tasks, including data collection, with minimal human intervention. Their ability to learn, adapt, and respond to changing environments makes them highly effective in handling complex and dynamic data collection needs [3].

Despite the promising benefits of intelligent data collection systems, several obstacles remain:

- Data Fragmentation: The proliferation of data sources, from IoT sensors to mobile applications, often leads to fragmented datasets that are difficult to integrate and analyze.
- Technical Expertise: Implementing intelligent data collection systems requires expertise in AI, machine learning, and big data analytics, which many organizations may lack.
- Data Privacy and Ethical Issues: As intelligent systems access vast amounts of personal and sensitive data, ensuring compliance with privacy regulations (e.g., GDPR) and addressing ethical concerns become more complex.

Here is a detailed breakdown of the method of using intelligent agents as intelligent means of data collection:

- Autonomous Data Gathering: Intelligent agents are equipped with decision-making capabilities that allow them to autonomously search, collect, and filter relevant data. They can be deployed across different environments, such as web pages, IoT networks, and social media platforms, to monitor and capture specific information. To apply some specific agent to the environment, sensors and actuators must be properly developed and integrated.

- Real-Time Data Processing: Once the data is collected, intelligent agents can process it in real-time, using embedded algorithms to analyze the information and draw meaningful conclusions.
- Scalability and Adaptability: One of the major advantages of using intelligent agents is their scalability. They can handle large volumes of data from multiple sources simultaneously, which is crucial in today's data-rich environments. Moreover, intelligent agents are adaptable—they can learn from their experiences and optimize their behavior over time, improving their efficiency in collecting and processing data.
- Integration with Machine Learning: Intelligent agents can be integrated with machine learning models to improve data collection processes. These models can help agents refine their data-gathering strategies by identifying patterns, predicting outcomes, and automatically adjusting their behavior to target more relevant data.
- Multi-Agent Systems: In some cases, multiple intelligent agents work together in a coordinated fashion to collect data from various sources. These agents communicate with one another, share information, and collaborate to achieve more comprehensive data collection.
- Privacy and Security Mechanisms: Given the sensitivity of the data collected, intelligent agents are often designed with robust privacy and security mechanisms to ensure data integrity and confidentiality.
- Decision Support and Automation: Intelligent agents not only collect and analyze data but also provide decision-support tools.

While intelligent agents offer many advantages in data collection, there is a very big challenge: the complexity of development – designing and deploying intelligent agents requires technical expertise in AI, machine learning, and software engineering. In addition to assumptions about the agent, many methods make assumptions about the environment within which the interaction takes place. Some common assumptions concern the order in which agents choose their actions (simultaneous or alternating moves), and the representation of actions and environment states (discrete, continuous, mixed). However, the most important assumptions usually concern the extent to which agents can observe what is happening in the environment [4].

The results of research. The implementation of the above methods provides the possibility to build a complex smart system that interacts with another environment where sensors get data and send them to the agent for further processing.

Implementation of intelligent agents with all capabilities can be slow by performance during execution and it can't be used for all the situations which appear in the environment. For such a case, the best option is to implement multilayered agents that perform some acts to the interacted environment based on complexity and priority.

Based on the complexity of execution, multilayered agents can have different responsibilities. Each agent can do different actions for the interacted environment:

- Simple reflex agents can manage scalability with linear logic.
- Model-based reflex agents can build models of the internal behavior of the interacted environment and show potential issues.
- Goal-based agents can measure the performance of the interacted environment and provide some ideas for improvements.
- Utility-based agents can analyze the evolution of the interacted environment and highlight issues with different historical behavior or legacy flows.
- Learning agents can provide intellectual changes and improvements to the interacted environment: smart preventive scalability, environment structure improvements, and different logical and unpredictable risk highlights.

Multilayered intelligent agents offer significant practical benefits in data collection by enhancing accuracy, scalability, and efficiency. They divide tasks across specialized layers, allowing for real-time processing, task specialization, and improved decision-making. This layered approach ensures resilience, fault tolerance, and better resource management, while also providing security and privacy protections. Additionally, these agents enable seamless integration across diverse platforms, making them highly adaptable and efficient for complex, large-scale data collection tasks.

Practical value. Intelligent agents bring substantial practical value to data collection by automating and optimizing processes, which significantly reduces manual effort, human error, and processing time. Their ability to operate autonomously enables real-time data collection and insights, which is crucial in industries like healthcare, finance, and e-commerce, where timely decisions can make a significant difference.

One of the key advantages of intelligent agents is their scalability. They can manage vast amounts of data from multiple sources simultaneously, providing organizations with a more comprehensive view of their operations. This is particularly beneficial in large-scale operations such as smart cities, logistics, or retail, where manual data collection would be inefficient or impractical.

Additionally, intelligent agents can improve data accuracy by filtering and validating collected information. They also adapt and learn over time through machine learning, enhancing their performance and ensuring the data remains relevant and precise. Moreover, these agents can be designed to ensure compliance with data privacy regulations, like GDPR, which is critical when handling sensitive information in fields such as healthcare and finance.

Overall, intelligent agents enhance operational efficiency, reduce costs, improve decision-making, and ensure high-quality data collection, offering significant practical value across a variety of industries.

Conclusions. Intelligent agents represent a powerful method for intelligent data collection, enabling real-time, scalable, and adaptive gathering of data across a wide range of applications. By leveraging AI and machine learning, these agents can autonomously handle the increasing complexity and volume of data in modern digital environments, while also addressing challenges like data privacy and security.

The research of the agents for data collection provides the knowledge and ideas for different kinds of data collection, data processing, and proper acts launch. The usage of ML and Data Mining provides complex capabilities for smart interaction with the interacted environment, as well as keeps historical data manipulation for modeling agents.

1. Russell, S. J., Norvig, P., & Davis, E. (2020). *Artificial intelligence: a modern approach* (4th ed.). Pearson.
2. Bringsjord, S., & Govindarajulu, N. S. (2018, July 12). Artificial Intelligence. In Edward N. Zalta (Ed.), *The Stanford Encyclopedia of Philosophy* (Summer 2020 Edition).
3. Doroshenko, A. V., & Savchuk, D. Yu. (2024). Research of data mining methods for classification of imbalanced data sets. Ukrainian Journal of Information Technology, 6(1), 48–57. https://doi.org/10.23939/ujit2024.01.048
4. Albrecht, S., & Stone, P. (2018). Autonomous Agents Modeling Other Agents: A Comprehensive Survey and Open Problems. *Artificial Intelligence*, 258, 66-95. https://doi.org/10.1016/j.artint.2018.01.002

Myroslav Mishchuk[1], Olena Pavliuk[2], Mykola Medykovskyy[3], Olena Litovska[4]
[1] *Lviv Polytechnic National University, Lviv, Ukraine, myroslav.mishchuk.mknus.2023@lpnu.ua*
[2] *Silesian University of Technology, Gliwice, Poland, olena.pavliuk@polsl.pl*
[3] *Lviv Polytechnic National University, Lviv, Ukraine, mykola.o.medykovskyi@lpnu.ua*
[4] *Lviv Polytechnic National University, Lviv, Ukraine, olena.litovska.kn.2021@lpnu.ua*

SMARTWATCH-BASED HUMAN STAFF ACTIVITY PREDICTION: THE USE CASE OF INTERNAL LOGISTICS SYSTEMS BASED ON AGVs

Introduction. Human Activity Recognition (HAR) has emerged as a fast-developing area of research, particularly with the era of Industry 4.0 and the ongoing transition to Industry 5.0. Initially, the focus of HAR was on basic activity classification, which has now been largely addressed. Today, the research has evolved to encompass more sophisticated challenges, such as complex human activity recognition, human activity analysis, and human activity prediction (HAP). In modern production environments, traditional production line systems such as belt conveyors or hangers are being increasingly replaced by Autonomous Guided Vehicles (AGVs). Unlike traditional systems, AGVs offer flexibility and adaptability in manufacturing processes. However, these systems require more complex coordination with human staff, making integrating advanced HAR and HAP systems critical [1]. Incorporated into Intellectual Enterprise Management (IEM) systems, these technologies allow for dynamic AGV routing and scheduling optimisation based on real-time human activity data. This integration can significantly boost the efficiency of production lines and internal logistics systems by adapting swiftly to changes in the workspace. Thus, this paper explores the application of smartwatch-based HAR systems within such contexts, proposing a new approach for HAP within an internal logistics system. The solutions based on the proposed approach can be incorporated as a part of the IEM systems to improve the efficiency of the production line.

Related works. Human activity analysis in industrial settings is approached in three major ways: camera-based, sensor-based, or a combination of both. While extensively applied for HAR, camera-based approaches have several disadvantages, including privacy concerns, the need for significant financial investments for comprehensive coverage, and constraints on staff mobility due to the necessity of remaining within camera view. These limitations are critical in dynamic sectors like agile manufacturing and internal logistics, where human staff mobility across large areas is common [2]. While multi-modal approaches based on wearable devices and cameras have been proposed, their application scope remains limited. A system presented by [7] requires operators to stay stationary and cameras to continuously monitor extensive areas, which is impractical when staff need to interact dynamically with AGVs. The wearable sensor-based approach avoids many challenges by eliminating the need for comprehensive facility coverage. However, current sensor-based solutions often demand expensive hardware and complex sensor fusion methods and may still restrict personnel mobility. The approach proposed in [6] employs a wearable system with IMU and body capacitance sensors to detect eleven worker activities. It uses a fusion of multimodal sensor data via convolutional neural networks and deep convolutional LSTM. The disadvantage is the need to develop dedicated sensors with 20 channels, which require specific equipment capable of continuous operation in industrial conditions. Moreover, the complexity of such systems significantly increases when the AGV is equipped with a collaborative robot, necessitating the fusion of diverse data sources [8]. The approach proposed in this study requires only a smartwatch - a device widely used in sports, which is cost-effective and compliant with many industrial internal policies. The only change from the smartwatch side is installing a dedicated application to collect sensor data. Depending on the architecture, this data can be processed directly on the device or transmitted to an edge server. Moreover, the stacked architecture of this system supports the implementation of Federated Learning (FL), enhancing data privacy and system adaptability.

Materials and methods. To support this research, a distributed smartwatch-based data collection system was developed, consisting of three main components: a smartwatch, a mobile application, and a cloud server. The smartwatch used is the Samsung Galaxy Watch 5, which operates on WearOS. The application's main purpose is to provide a user interface for controlling the experiment flow, gathering sensor data, and sending it to the cloud. The cloud server is based on the Platform as a Service (PaaS) model and is used to store and provide the gathered data for analysis. The signals are collected from the tri-axial accelerometer and gyroscope at a frequency of 100 Hz and then transmitted and stored in the form of the "dataframe" data structure. Each dataframe contains 3 seconds of the recorded 6-channel signals with information about activity labels and start and stop timestamps. The dataset for this study was compiled using inputs from five human staff representatives who performed continuous repetitions of one of two predefined sequences of activities designed to simulate typical tasks in an internal logistics environment. The first sequence includes the following activities: sitting at point 1, transitioning to standing, standing, walking to point 2, executing a 90-degree rotation, walking to point 3, standing, transitioning back to sitting, and sitting. The activities are then repeated in reverse order to return to the starting position. The second sequence consists of sitting at point 1, transitioning to standing, standing, walking to point 2, executing a 180-degree rotation in either direction, walking back to point 1, standing, transitioning back to sitting, and resuming a seated position. Throughout these experimental sessions, participants submitted information about specific activities and sequences they were performing. The accumulated dataset encompasses 3.28 hours of recorded signals. The proposed approach utilises a stacked architecture containing a classifier for activity recognition from sensor signals and a predictor for forecasting subsequent activities. The preprocessing stage consists of 5 stages. First, dataframes outside the temporal boundaries of the experiments are discarded. Second, interruptions are identified using a 500 ms temporal delta criterion. Third, uninterrupted sequences are segmented into fixed-size, 60-second subsequences (comprising 20 dataframes each). Fourth, a 50% overlap is applied, increasing the sample count. Finally, the data is shuffled and split into three parts, allocating 40% for training both the base classifier and predictor, the other 40% for fine-tuning the predictor and validating the base classifier, and the remaining 20% for testing both classifiers. This strategy prioritises extensive training for the predictor, though it may overlook some errors of the base classifier on previously unseen data. The model proposed by [4] is used as the base classifier. It leverages the DenceNet121 architecture and is specifically designed for HAR tasks by pre-training on an extensive HAR dataset. Additionally, it employs Continuous Wavelet Transform to enhance productivity and mitigate overfitting and underfitting problems [5]. The parameters identified as optimal in [3] use a Morlet wavelet with 256 scales and 136 of the model's layers frozen to enhance training. For activity prediction, a range of models were employed, including LSTM, BiLSTM, GRU, and BiGRU, in both multi-layer and single-layer configurations. These models are widely used for time series classification and prediction. The predictor utilises the outputs from the top layer of the base classifier, processing a sequence of 19 predictions to generate a probability-like distribution of the next likely activity. This setup enables real-time prediction and allows a distributed architecture, suitable for FL applications.

Results. The base activity classifier achieves the performance metrics as follows: accuracy at 90.90%, precision at 91.33%, recall at 90.69%, AUC at 97.26%, and F1-score at 91.01%. These results indicate sufficient performance, particularly considering the limited training data and potential activity overlaps within the 3-second dataframes. Notably, the F1-score, resilient to dataset imbalance, confirms the classifier's impartiality across various activities. Subsequently, activity labels across the dataset were redefined based on predictive outcomes from the top-layer neurons of the base classifier. The achieved predictors' performances are provided in Table 1. The best-performing model, Multi-layer BiLSTM, achieved the F1-score of 63.49%. Despite the relatively low prediction performance, we consider it sufficient given the constraints of a small dataset, significant dataset imbalance, and potential activity overlap within a dataframe. Future enhancements are expected to significantly improve outcomes by expanding the dataset size, employing more sophisticated prediction models (e.g., autoencoders and heterogeneous models), and enhancing the base classifier's performance. Future work will refine prediction accuracy using these strategies and integrate the improved predictive system within a real production environment. This will involve integrating the proposed approach with the IEM system and leveraging FL to distribute the activity classifier and predictor functions.

Table 1. The achieved predictors' performances

Predictor	Accuracy, %	Precision, %	Recall, %	AUC, %	F1-score, %
Single-layer LSTM	66.67	73.47	50.00	82.71	59.50
Multi-layer LSTM	69.44	71.15	51.39	83.44	59.68
Single-layer BiLSTM	65.28	82.86	40.28	80.84	54.21
Multi-layer BiLSTM	65.28	74.07	55.56	82.93	*63.49*
Single-layer GRU	66.67	72.92	48.61	82.75	58.33
Multi-layer GRU	68.06	80.43	51.39	85.41	62.711
Single-layer BiGRU	63.89	72.73	55.56	81.48	63.00
Multi-layer BiGRU	68.06	72.22	54.17	83.90	61.90

Conclusions. In conclusion, this study proposes a smartwatch-based approach for human activity prediction within internal logistics systems that utilise AGVs. A HAR-specific, DenseNet121-based model was employed for activity recognition, achieving the F1-score of 91.01%. For activity prediction, LSTM, BiLSTM, GRU, and BiGRU architectures in both multi-layer and single-layer configurations were tested, with the best-performing model based on the Multi-layer BiLSTM achieved an F1-score of 63.49%. Despite the relatively low prediction performance, we consider it adequate given the dataset limitations. Future works will focus on refining prediction accuracy by expanding the dataset and employing more sophisticated prediction models, as well as integrating the proposed predictive system into a real production setting. The proposed approach offers potential enhancements to the efficiency of production lines by integrating with Intellectual Enterprise Management systems.

1. Cupek, R., Drewniak, M., Fojcik, M., Kyrkjebø, E., Lin, J. C.-W., Mrozek, D., Øvsthus, K., & Ziebinski, A. (2020). Autonomous Guided Vehicles for Smart Industries – The State-of-the-Art and Research Challenges. In Computational Science – ICCS 2020. Springer International Publishing, 330–343. https://doi.org/10.1007/978-3-030-50426-7_25
2. Cupek, R., Lin, J. C.-W., & Syu, J. H. (2022). Automated Guided Vehicles Challenges for Artificial Intelligence. 2022 IEEE International Conference on Big Data (Big Data), 6281–6289. https://doi.org/10.1109/BigData55660.2022.10021117
3. Pavliuk, O., & Mishchuk, M. (2022). A novel Deep-Learning model for Human Activity Recognition based on Continuous Wavelet Transform. CEUR Workshop Proceedings, 3302, 236–245. https://ceur-ws.org/Vol-3302/paper14.pdf
4. Pavliuk, O., Mishchuk, M., & Strauss, C. (2023). Transfer Learning Approach for human activity recognition based on Continuous Wavelet transform. Algorithms, 16(2), 77. https://doi.org/10.3390/a16020077
5. Prots'ko, I., & Mishchuk, M. (2021). Block-Cyclic Structuring of the Basis of Fourier Transforms Based on Cyclic Substitution. Cybernetics and Systems Analysis, 57(6), 1008–1016. https://doi.org/10.1007/s10559-021-00426-x
6. Suh, S., Rey, V. F., Bian, S., Huang, Y., Rožanec, J. M., Ghinani, H. T., Zhou, B., & Lukowicz, P. (2023, August 7). Worker activity recognition in manufacturing line using near-body electric field. arXiv.org. https://arxiv.org/abs/2308.03514
7. Tao, W., Leu, M. C., & Yin, Z. (2020). Multi-modal recognition of worker activity for human-centered intelligent manufacturing. Engineering Applications of Artificial Intelligence, 95, 103868. https://doi.org/10.1016/j.engappai.2020.103868
8. Ziebinski, A., Mrozek, D., Cupek, R., Grzechca, D., Fojcik, M., Drewniak, M., Kyrkjebø, E., Lin, J. C., Øvsthus, K., & Biernacki, P. (2021). Challenges Associated with Sensors and Data Fusion for AGV-Driven Smart Manufacturing. In Lecture notes in computer science, 595–608. https://doi.org/10.1007/978-3-030-77970-2_45.

Olga Narushynska[1]
[1] *Lviv Polytechnic National University, Lviv, Ukraine, narushynska@gmail.com*

THE ROLE OF BYTE PAIR ENCODING IN PREPROCESSING TEXT DATA FOR HIERARCHICAL CLASSIFICATION

Introduction. We live in a time when the economy is becoming increasingly globalized, and trade is one of the most important sectors. The range of goods is highly diverse, which creates a need for hierarchical classification. This allows for the creation of multi-level structures that efficiently organize goods into categories and subcategories. Hierarchical classification of goods is essential for managing logistics, sales, and marketing in various industries [1]. With the growing complexity and volume of data related to consumers and their purchases, it is increasingly challenging to develop effective algorithms for automatically classifying goods according to their hierarchical structure. Doing this manually for a large number of products is not only difficult but also time- and resource-intensive. Therefore, automated approaches to hierarchical classification are necessary to maintain efficient logistics and business processes. Machine learning is a powerful tool for building models that can classify goods based on their characteristics and hierarchical structure.

The focus of this research is to improve the preprocessing of textual data for hierarchical classification by integrating Byte Pair Encoding (BPE) [3] with BERT embeddings [7]. BPE allows for more effective segmentation of text into subword units, which can better capture the semantics of abbreviated or rare terms. This preprocessing step enhances the performance of the classification model without requiring retraining of the BERT model itself.

The object of this research is the process of text data preprocessing and hierarchical classification in natural language processing systems.

The subject of this research includes methods and techniques for improving text data preprocessing, particularly by integrating BPE with BERT embeddings to enhance the accuracy of hierarchical classification models [6].

Materials and methods. This work aims to develop a text preprocessing approach that combines Byte Pair Encoding (BPE) with BERT embeddings to improve the accuracy of hierarchical classification tasks. The approach is designed to effectively handle short texts with abbreviations without requiring retraining of the base model, thereby offering a scalable and efficient solution for enhancing text classification performance in various industries.

The relevance of this study lies in the fact that the accuracy and efficiency of goods classification are critical factors in many industries, including retail, logistics, and marketing. The methods described in this research can be applied to enhance the efficiency of business processes and improve customer satisfaction. Considering the development of the global economy and trade, the hierarchical classification of goods is becoming even more important [1]. It allows for standardizing the classification of goods, providing greater accuracy and uniformity of product data worldwide. This is especially crucial for cross-border trade operations and international cooperation between companies.

In this study, we apply BPE [6] as a preprocessing step to enhance the input representation for hierarchical classification models. BERT embeddings were selected for their robust contextualized representations of words and phrases, but they can still struggle with the tokenization of rare or abbreviated terms. By applying BPE before generating BERT embeddings [3], the text data is divided into subword units, which allows the model to handle previously unseen words or abbreviations better.

To address the issue of hierarchical classification, this research utilized the GS1 Global Product Classification (GPC) system, accessible via https://gpc-browser.gs1.org/. This hierarchy was selected for its widespread usage across various industries and its comprehensive class and subclass structure, which facilitated a thorough analysis of different approaches to hierarchical classification [2]. The GS1 GPC structure consists of four levels: segment, family, class, and subclass. We focused on these categories to concentrate on popular and widely used products. The experiment utilized two datasets: the first dataset comprised product descriptions from a company supplying the restaurant industry, manually labeled according to a GS1 Global Product Classification (GPC) taxonomy. The second dataset was sourced from the website https://www.directionsforme.org/, containing product descriptions aligned with the same hierarchical classification system. The BPE-enhanced embeddings were compared to traditional preprocessing methods to evaluate improvements in accuracy [4].

Results. Table 1 presents the performance metrics of machine learning models (XGBoost, PerNode, PerLevel) applied to hierarchical classification. PerNode and PerLevel are hierarchical classification models often used to address problems where the data has a tree-like structure [2]. Both methods aim to improve the accuracy of hierarchical classification by leveraging the structural relationships in the data, but they differ in terms of the localization of their decision-making process. Various text preprocessing methods were evaluated, including Byte Pair Encoding (BPE), BERT embeddings, and word2vec. Models using BPE in preprocessing demonstrated superior performance, particularly in cases involving abbreviated product descriptions.

Table 1. Performance of various models with different preprocessing techniques

Model	Dataset	Runtime, sec	Brick_Accuracy	Brick_Precision	Brick_F1 Score	Brick_Recall
XGBoost	bert_concat_BPE	842	0,717	0,724	0,716	0,717
XGBoost	bert_embeddings	1806	0,699	0,710	0,698	0,699
XGBoost	word2vec	1024	0,678	0,678	0,674	0,678
XGBoost	one_hot_result	25	0,064	0,004	0,008	0,064
XGBoost	tfidf_result	83	0,064	0,004	0,008	0,064
XGBoost	BPE	18	0,064	0,004	0,008	0,064
PerNode	bert_concat_BPE	193	0,641	0,691	0,635	0,641
PerNode	bert_embeddings	463	0,637	0,687	0,629	0,637
PerNode	word2vec	132	0,628	0,687	0,621	0,628
PerNode	tfidf_result	11	0,074	0,005	0,010	0,074
PerNode	one_hot_result	12	0,074	0,005	0,010	0,074
PerNode	BPE	21	0,074	0,005	0,010	0,074
PerLevel	bert_concat_BPE	357	0,583	0,633	0,571	0,583
PerLevel	bert_embeddings	44	0,579	0,636	0,566	0,579
PerLevel	word2vec	30	0,547	0,584	0,528	0,547
PerLevel	one_hot_result	5	0,074	0,005	0,010	0,074
PerLevel	tfidf_result	6	0,074	0,005	0,010	0,074
PerLevel	BPE	8	0,074	0,005	0,010	0,074

Through the implementation of machine learning models goods were classified according to their hierarchical structures with a high level of accuracy [5]. These approaches significantly reduced manual effort and improved the speed of data processing. The integration of BPE in text preprocessing resulted in significant improvements in the accuracy of the hierarchical classification model. Specifically, the BPE-enhanced BERT [7] embeddings led to a 2% increase in classification accuracy compared to traditional BERT embeddings. This was particularly evident in handling short texts and descriptions with abbreviations, where traditional embeddings struggled to capture the full meaning.

Adding Byte Pair Encoding (BPE) as a preprocessing step in text data processing offers several key advantages, particularly in hierarchical classification tasks:

- Handling Rare and Abbreviated Terms: BPE is highly effective at dealing with rare words or abbreviations that may not appear frequently in the training data. By breaking down words into subword units, BPE ensures that even unseen or uncommon words can be processed meaningfully by the model. This is particularly important for product descriptions where abbreviations and domain-specific terms are common.
- Efficient Use of Vocabulary: BPE helps limit the size of the model's vocabulary by encoding frequently occurring subwords instead of expanding the vocabulary with every new word form. This

reduces memory requirements while still maintaining a high level of expressiveness, making the model more efficient in terms of both computation and storage.

- Compatibility with Pre-trained Models: Since BPE works on the preprocessing level, it can be integrated with pre-trained models like BERT without modifying the model architecture itself [7]. This means improvements in text representation can be achieved without the need for retraining, saving time and computational resources.
- Boosting Accuracy for Hierarchical Classification: In tasks like hierarchical classification, where the specificity of terms is crucial, BPE ensures that even products with niche or abbreviated descriptions are classified accurately. The ability to represent abbreviations as meaningful subword sequences improves the model's ability to correctly categorize products based on limited textual input.

Conclusions. Adding BPE enhances the model's ability to process and classify text by improving tokenization, handling rare terms, and optimizing vocabulary usage, all without the need for retraining large models like BERT. This leads to improved classification accuracy, especially in specialized tasks like hierarchical classification of goods.

The practical significance of this research is its application in improving product classification accuracy in various industries, such as retail, logistics, and e-commerce. The proposed method, which combines BPE with BERT embeddings, can be easily integrated into existing data pipelines, offering a low-cost solution to enhance the accuracy of hierarchical classification models without retraining. This approach is particularly valuable for businesses dealing with large, diverse product catalogs, where accurate classification is critical for inventory management and customer satisfaction.

This study presents a novel approach to text data preprocessing by integrating Byte Pair Encoding (BPE) with BERT embeddings for hierarchical classification tasks. The results indicate that this preprocessing method significantly improves model accuracy, especially for short texts with abbreviations, without the need for retraining the underlying model. The method provides a scalable solution for improving classification in industries that rely on large datasets of product descriptions. Future research could explore the application of BPE in other natural language processing tasks and further enhance its integration with multi-lingual models.

1. Borges, H. B., Silla Jr., C. N., & Nievola, J. C. (2013). An evaluation of global-model hierarchical classification algorithms for hierarchical classification problems with single path of labels. Computers & Mathematics with Applications, 66 (10), 1975-1987. https://doi.org/10.1016/j.camwa.2013.06.027
2. Gerrit J.J. van den Burg, Alfred O. Hero (2017). Fast meta-learning for adaptive hierarchical classifier design. arXiv preprint arXiv:1711.03512. https://arxiv.org/pdf/1711.03512
3. Lian, H., Xiong, Y., Niu, J., Mo, S., Su, Z., Lin, Z., Liu, P., Chen, H., & Ding, G. (2024). Scaffold-BPE: Enhancing Byte Pair Encoding with simple and effective scaffold token removal. arXiv preprint arXiv:2404.17808. https://arxiv.org/abs/2404.17808
4. Miranda, F. M., Köhnecke, N., & Renard, B. Y. (2023). HiClass: A Python library for local hierarchical classification compatible with scikit-learn. Journal of Machine Learning Research, 24, 1-17. https://arxiv.org/abs/2112.06560
5. Rezende, P. M., Xavier, J. S., Ascher, D. B., Fernandes, G. R., & Pires, D. E. V. (2022). Evaluating hierarchical machine learning approaches to classify biological databases. Briefings in Bioinformatics, 23(4), bbac216. https://doi.org/10.1093/bib/bbac216
6. Tahmid, M. T., Shahgir, H. S., Mahbub, S., Dong, Y., & Bayzid, M. S. (2024). BiRNA-BERT allows efficient RNA language modeling with adaptive tokenization. *bioRxiv*. https://doi.org/10.1101/2024.07.02.601703
7. Talebi, S., Tong, E., Li, A., Yamin, G., Zaharchuk, G., & Mofrad, M. R. K. (2024). Exploring the performance and explainability of fine-tuned BERT models for neuroradiology protocol assignment. BMC Medical Informatics and Decision Making, 24(40). https://doi.org/10.1186/s12911-024-02444-z.

Oleksandr Ohlashennyi[1], Olha Narushynska[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, olexandrohlashennyi@gmail.com
[2]Lviv Polytechnic National University, Lviv, Ukraine, narushynska@gmail.com

TECHNICAL VISION SYSTEM DEVELOPMENT FOR MULTICOPTER DRONE IDENTIFICATION

Introduction. The rapid development and widespread use of unmanned aerial vehicles (UAVs), particularly multicopters, have become prevalent across various fields, from military operations to civilian applications. Their deployment in military conflicts for reconnaissance, bombings, and kamikaze drones poses significant threats to civilian populations and critical infrastructure. In wartime, the effective detection and neutralization of UAVs are vital for ensuring national security and defense.

The object of design is the process of image processing and object identification in technical vision systems.

The subject of design includes methods, models, and hardware of the technical vision system for detecting unmanned aerial vehicles of the multicopter type [1].

This work aims to develop a system for real-time detection of multicopter UAVs using modern image processing and machine learning methods. The system shall autonomously and accurately identify drones under various conditions, requiring comprehensive visual analysis and adaptability to changing environments. It is designed to be an essential tool for defense operations, infrastructure protection, and other organizations that need robust defense against potential aerial threats.

Key tasks involve:

- Developing technical vision algorithms that employ machine learning for the automatic recognition of drones.
- Preparing diverse datasets for training and validation.
- Selecting optimal components for cost-effective operation.
- Developing a hardware solution that supports autonomous function.

Additionally, the system will feature a camera rotation mechanism for precise UAV tracking and integrate with response systems to activate protective measures upon drone detection. Careful testing, including actual drone trials, will be conducted to refine the system continuously.

The relevance of this work is underscored by the growing necessity for effective UAV detection systems, particularly in modern warfare, where drones are extensively used. The system's practical significance lies in its ability to monitor airspace and detect threats, offering enhanced protection in combat situations and safeguarding critical infrastructure. In the face of advancing technology and increasing UAV threats, developing a technical vision system for drone identification is crucial for improving security and protecting vital assets and lives.

Materials and methods. The system includes:

- The technical vision system developed in this work in Python using the YOLOv8 convolutional neural network from the ultralytics library, trained on a dataset of 18,976 images of multicopter drones. It runs on a Nvidia Jetson Nano single-board computer, processing camera images to detect UAVs [2, 3].
- The electric signal generation module generates an output signal on the Nvidia Jetson Nano when a UAV is detected, enabling integration with other UAV threat response systems.
- The rotating platform expands the scanning area and enables UAV tracking. It consists of a camera mounted on stepper motors controlled by an STM32F103 microcontroller programmed in C using HAL libraries [4].
- Communication between the technical vision system on the Nvidia Jetson Nano and the rotating platform is established via UART, allowing Jetson Nano to send rotation commands to STM32F103 [5].
- The system body securely houses the rotating platform and all components. It was created using 3D modeling and 3D printing technologies.

Results. The system was successfully developed to detect multicopter UAVs using a camera and a technical vision system, generate an external electrical signal upon detection, and track the detected drone by directing the camera toward it.

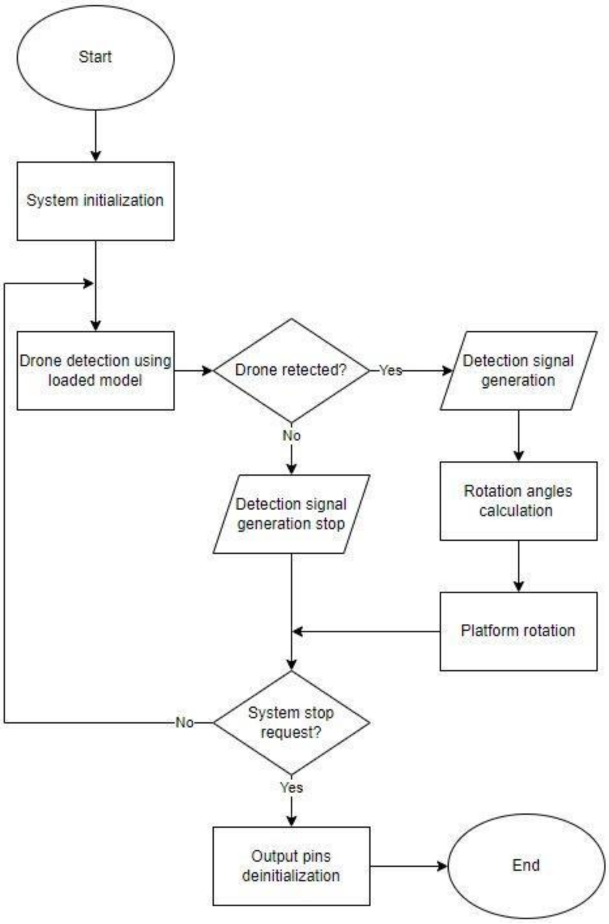


Fig. 1. System`s general algorithm

Figure 1 illustrates the system`s principle of work: from the initialization all the way to the actual drone detection.

Aside from the technical vision system, the rotating platform was created from scratch with an STM32F103 microcontroller and 28byj-48 stepper motors, and a 3D model was designed to host all the system's components and implement the rotating mechanism.

Fig. 2. Assembled device

Figure 2 illustrates the assembled ready-to-use device, which houses the Nvidia Jetson Nano, STM32F103, ULN2003 stepper motor drivers, and the rotating platform with a camera, which is driven by 28byj-48 stepper motors.

To verify the system's accuracy and reliability, the trained convolutional network YOLOv8 was validated, and the results are displayed in Figure 3.

```
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100%|██████████| 140/140
                   all       2230       2437      0.964      0.947      0.971       0.69
Speed: 2.5ms preprocess, 127.9ms inference, 0.0ms loss, 0.5ms postprocess per image
```

Fig. 3. YOLOv8 model validation result

The results of the validation metrics include:

- Precision (P): 0.964 – This indicates that when the model predicts the presence of a drone, the prediction is correct about 96.4% of the time.
- Recall (R): 0.947 – This shows that the model successfully identifies 94.7% of all drones in the dataset [6].

Conclusion. This work successfully developed and tested a technical vision system for identifying multicopter-type UAVs, demonstrating its critical role in addressing the growing use of drones in warfare. The system enhances the ability to respond swiftly to aerial threats, protecting lives, equipment, and essential infrastructure.

The conducted analysis and experiments confirmed the system's high effectiveness in drone detection, making it suitable for military use, as well as for safeguarding critical infrastructure, airports, and private properties.

A key future direction is the system's integration with response and counter-UAV technologies, enabling proactive alerts and automatic countermeasures through electronic warfare or air defense systems.

Future improvements will focus on increasing response speed and detection range and integrating additional sensors to enhance functionality and detection accuracy.

Overall, the developed system represents a significant advancement in security technology, contributing to national security by applying cutting-edge technical vision technologies.

1. Shandilya, S. K., Srivastav, A., Yemets, K., Datta, A., & Nagar, A. K. (2023). YOLO-based segmented dataset for drone vs. bird detection for deep and machine learning algorithms. *Data in Brief*, 50, 109355. https://doi.org/10.1016/j.dib.2023.109355
2. Redmon, J., & Divvala, S. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (CVPR) (pp. 779-788).
3. Valladares, S., & Toscano, M. (2020). Performance evaluation of the NVIDIA Jetson Nano through a real-time machine learning application. *Advances in Intelligent Systems and Computing.* https://doi.org/10.1007/978-3-030-41620-7_28
4. STMicroelectronics. (n.d.). *Description of STM32F1 HAL and low-layer drivers.* Retrieved from https://www.st.com/resource/en/user_manual/um1850-description-of-stm32f1-hal-and-lowlayer-drivers-stmicroelectronics.pdf
5. Peña, E., & Legaspi, M. G. (n.d.). UART: A hardware communication protocol understanding universal asynchronous receiver/transmitter. *Analog Devices.* Retrieved from https://www.analog.com/en/resources/analog-dialogue/articles/uart-a-hardware-communication-protocol.html
6. Goutte, C., & Gaussier, E. (2017). A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. *Computational Linguistics, 30*(2), 249-257.

Sofiia Oleskevych[1], Mariana Seneta[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, sofiia.oleskevych.mknus.2023@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, mariana.y.seneta@lpnu.ua

APPLYING ARTIFICIAL INTELLIGENCE FOR DISEASE DETECTION IN DENTAL DOMAIN

Introduction. Technological advancements in healthcare have revolutionized medical practices, and dentistry is no exception. The increasing adoption of Artificial Intelligence (AI) in various fields is reshaping dental care, from diagnosis to treatment planning.

The integration of Artificial Intelligence (AI) into diagnostics and treatment planning in dentistry is rapidly gaining traction. With the ability to analyze complex datasets, recognize patterns, and assist in decision-making. As technology evolves at a rapid pace, keeping up with these trends is not only beneficial but necessary to stay relevant in today's dental practices. There are many ways in which AI could help in dentistry:

1. X-rays analysis.
2. Disease detection with CNNs.
3. Treatment planning and predicting.
4. Dental implantation with AI assistance [1].

This article focuses on how neural networks can detect dental anomalies. We will compare various models, such as YOLOv5, U-Net, Faster R-CNN [2], and DeepLabV3+ to determine the most effective solution for this specific task.

Materials and methods. CNNs, also known as convolutional neural networks, are deep learning models specialized in excelling at tasks related to recognizing images. When it comes to dentistry, CNNs are specifically trained using extensive sets of labeled dental images, including X-rays and medical images. During the training process, the network becomes adept at identifying particular features linked to dental conditions, such as cavities, fractures, or bone loss, and can subsequently apply this knowledge to detect similar features in new, previously unseen images. The key advantage of CNNs is their ability to learn hierarchical patterns [3]. In the dental sphere, this means that the network can start by recognizing basic shapes and contours of teeth and gums, and then progressively detect more complex patterns, such as carious lesions or irregularities in the bone structure. This layered approach allows for extremely precise identification of dental issues, even in the early stages.

YOLOv5 is an advanced real-time object detection model known for its speed and accuracy. YOLOv5 processes the entire image in one pass and predicts both the bounding boxes and class labels for objects in real time. It is highly optimized for detecting multiple objects at once, which makes it ideal for tasks like detecting dental anomalies (e.g., cavities, crowns, fillings). YOLOv5 has been widely adopted due to its superior speed and lower computational requirements compared to earlier versions like YOLOv3 and YOLOv4, and it’s especially suitable for mobile or cloud-based deployments [4].

U-Net is a convolutional neural network architecture initially developed for biomedical image segmentation. Unlike object detection models like YOLO, U-Net focuses on segmenting images into meaningful areas, which makes it ideal for identifying specific regions in medical images. In dentistry, U-Net can be used to highlight regions affected by dental diseases, such as decayed areas or bone loss. Its encoder-decoder architecture is particularly useful for creating detailed masks of anomalies in complex images, offering highly accurate segmentation of dental structures.

Faster R-CNN is an evolution of the R-CNN series and is known for its accuracy in object detection. Faster R-CNN improves upon previous models by combining region proposal networks (RPN) and object detection into a single, faster pipeline. This model works well for precise detection tasks, such as identifying fractures or other dental anomalies in X-rays or CT scans [5]. However, it is computationally intensive and slower compared to YOLOv5, which can be a limitation in real-time applications. Comparative characteristics of all previously mentioned models are listed in Table 1.

Table 1. Comparative analysis of YOLO, U-Net, Faster R-CNN

Aspect	YOLOv5	U-Net	Faster R-CNN
Speed	Very fast (real-time)	Moderate (slower than Yolov5)	Slow (multistage)
Accuracy	High	High (pixel level accuracy)	Very high (Precise detection of small anomalies)
Architecture	Single-stage detection (backbone + neck + head)	Encoder-decoder with skip connection	Two-stage (RPN for region proposals + object detection)
Main feature	Object detection	Pixel segmentation, detection	Object detection
Real-time capabilities	Yes (highly optimized)	No	No
Use cases	Detecting dental issues (cavities, crowns, etc.)	Segmenting areas of interest: decayed areas, gums, bone loss	Detecting small fractures, early-stage diseases
Handling small objects	Good	Not optimized for small object detection	Excellent

For real-time detection of dental anomalies, YOLOv5 is the most suitable model, offering an optimal balance between speed and accuracy. Faster R-CNN is ideal for high-precision tasks in offline applications, while U-Net is best for tasks requiring detailed segmentation. For general dental diagnostics in real-time, YOLOv5 remains the most practical and efficient choice [7].

Results. The authors of the paper utilized two main architectures of the Faster Region-based Convolutional Neural Network (Faster R-CNN) for detecting dental caries: ResNet-50 and ResNet-101 [6]. Describing the architectures of these two models, ResNet-50 has 50 layers in deep and region proposal network (RPN) proposes regions for possible dental anomalies, and the RoI pooling layer reshapes them into fixed-size grids for further classification. While ResNet-101 has 101 layers deep, and due to its increased depth, ResNet-101 captures more intricate details, making it more powerful for object detection tasks but slower to train. Both ResNet modes were trained on a dataset of 81 dental caries images, whereas the test dataset contained 14 images. The ResNet-101 model trained with the Adam optimizer produced a lower mAP of 0.192, while the model using the momentum optimizer achieved a significantly lower mAP of 0.004, suggesting that the choice of optimizer greatly affects performance. The ResNet-50 trained using the Adam optimizer achieved a mean average precision (mAP) of 0.213. To summarize, the models were evaluated using mean average precision (mAP) as a key performance metric, with ResNet-50 paired with Adam optimizer showing the best performance among the models tested.

The authors of research used a YOLO model to detect dental cavity [4]. YOLO follows a single-stage object detection framework, which means that the entire image is processed in one pass. The figure 1 represents the backbone architecture of YOLO.

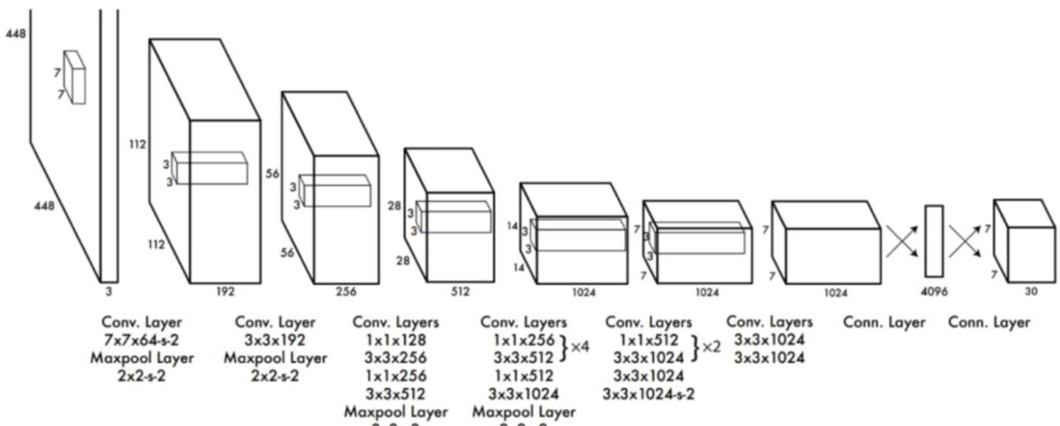


Fig. 1. Yolo architecture has 24 convolutional layers followed by 2 fully connected layers. Alternating 1x1 convolutional layers reduces the feature space from the preceding layers

The authors used YOLOv3, the main improvements of which are an increased amount of 53 convolutional layers and the ability to achieve state-of-the-art results. The dataset was borrowed from an Indian college called SMBT DENTAL COLLEGE & HOSPITAL and consists of up to 500 dental X-rays. In this research, the input images were resized to 416 x 416 pixels, and a confidence threshold of 0.3 was applied to retain only predictions with a certainty above 30%. Also, an Adam optimizer was used, with an initial learning rate of 0.001, and a dynamic scheduler was used to facilitate stable and efficient training over 100 epochs. Finally, the authors managed to achieve 81% accuracy in detecting dental cavities.

Conclusions. Developing a system for detecting dental anomalies [8] such as cavities, crowns, fillings, and sealed canals from dental X-rays or intraoral images, YOLO is the best choice, as it is designed for real-time object detection, meaning it processes the entire image in one pass, which drastically reduces the time needed for inference to milliseconds, making the user experience seamless. A key distinction of our approach is that, unlike most existing studies, which primarily focus on identifying a single class of issues (such as cavities), our work aims to detect multiple classes. Furthermore, many prior studies have relied on relatively small datasets, which inherently limit the robustness of the results. Fine-tuning the YOLO model on a specific dataset will allow the desired result to be achieved.

1. Suárez, A., Jiménez, J., & Llorente de Pedro, M. (2023). Beyond the Scalpel: Assessing ChatGPT's potential as an auxiliary intelligent virtual assistant in oral surgery. *Computational and Structural Biotechnology Journal*, *24*, 46–52. https://doi.org/10.1016/j.csbj.2023.11.058
2. Gandhi, R. (2018). R-CNN, Fast R-CNN, Faster R-CNN, YOLO - Object detection Algorithms. *Medium*. https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e
3. Mishra, M. (2020). Convolutional Neural Networks, Explained. *Medium*. https://towardsdatascience.com/convolutional-neural-networks-exolained-9cc-5188c4939
4. Kohar, R., & Sonavane, A. (2022). Dental Cavity Detection Using YOLO. In *Lecture Notes on Data Engineering and Communications Technologies, 91*, 141–152.
5. DigitalOcean. (n.d.). Faster R-CNN Explained for Object Detection Tasks. *DigitalOcean | Cloud Infrastructure for Developers.* https://www.digitalocean.com/community/tutorials/faster-r-cnn-explained-object-detection
6. Lanyak, A. C. F., Prasetiadi, A., Widodo, H. B., Ghani, M. H., & Athallah, A. (2024). Dental caries detection using faster region-based convolutional neural network with residual network. *IAES International Journal of Artificial Intelligence (IJ-AI)*, *13*(2), 2027. https://doi.org/10.11591/ijai.v13.i2.pp2027-2035 .
7. Aggarwal, A. (2020). YOLO Explained. *Medium*. https://medium.com/analytics-vidhya/yolo-explained-5b6f4564f31
8. Muresan, M. P., Barbura, A. R., & Nedevschi, S. (2020). Teeth Detection and Dental Problem Classification in Panoramic X-Ray Images using Deep Learning and Image Processing Techniques. In *2020 IEEE 16th International Conference on Intelligent Computer Communication and Processing (ICCP)*. IEEE. https://doi.org/10.1109/iccp51029.2020.9266244.

Andrii Palii[1], Iryna Kazymyra[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, andrii.palii.mknus.2023@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, iryna.y.kazymyra@lpnu.ua

INFORMATION SYSTEM FOR ORGANIZING TRAINING WITH THE USE OF AI TOOLS

Relevance of the research. The modern world is undergoing global change and challenges affecting various aspects of human life. One such challenge is the military aggression against Ukraine, which has caused significant losses among the population, both physical and moral. A large number of people have suffered from the hostilities, receiving injuries of varying severity, which has significantly changed their usual way of life and physical activity. In this situation, the issue of restoring health through physical exercise becomes particularly relevant but at the same time requires an individual approach to each victim.

Given the impact of injuries on physical and mental health, it is important to adapt the training process so that it is safe and effective. Exercises should be adapted to the limitations that exist. For example, for people with knee problems, reducing the load on the joints and including exercises that do not involve significant knee bending can be helpful. Alternative types of training can be used, such as water exercises, which reduce the load on the joints. It is important to gradually increase the intensity and complexity of training to avoid overuse and injury. Proper exercise technique is critical for avoiding additional injuries.

According to a study published in [1–3], exercise adaptation for people with injuries includes modification of the load, use of special techniques, and support for psychological support. The study shows that people who receive personalized support in the rehabilitation process have better physical and mental results. Another study published in the International Journal of Sports Medicine indicates that exercise can help manage symptoms of anxiety and depression symptoms if they are adapted to the patient's capabilities and limitations. As a result, the adaptation of training for people with injuries requires consideration of both physical and mental aspects. The right support, personalized exercises, and proper psychological support can significantly improve the overall condition and quality of life for people with medical conditions.

Methods and materials. Many existing AI-powered training systems, such as Peloton, Fitbit Premium, and WHOOP, effectively address offline fitness tracking and analysis. These platforms typically offer comprehensive post-workout insights, including detailed metrics and progress tracking. However, their reliance on offline data processing limits their ability to provide real-time feedback and adjustments during workouts.

1. Peloton – interactive fitness platform that provides users with access to a variety of workouts (including cycling, running, yoga, and strength training) with live or on-demand instructors.
2. Fitbit Premium – a service from Fitbit, a fitness tracker manufacturer, that offers personalized recommendations for physical activity, sleep, stress, and nutrition based on data collected from wearable devices.
3. WHOOP – a system for analyzing workouts and recovery that uses wearable devices to collect data on a user's activity, sleep, and recovery.

While there are many AI-powered systems for organizing training or rehabilitation, each has its unique features [4]. Most focus on personalizing workout programs and use data from wearable devices or motion analysis to provide recommendations. Many similar programs only work with specialized smart trackers, and functionality increases depending on the subscription, with prices that seem somewhat absurd.

Adapting workouts for people with physical injuries or special needs is a rapidly developing area but still has significant potential for improvement. A system that takes into account the Ukrainian context, particularly the rehabilitation of war-wounded individuals, could occupy an important niche in this market.

Results of research. An information system for organizing workouts using artificial intelligence (AI) tools involves the integration of modern technologies to automate and improve the process of physical training. The main goal of such a system is to create individualized training plans based on user data, such as age, weight, physical condition, injuries or limitations, as well as personal goals (weight loss, muscle gain, etc.).

Developing such a system involves several important steps. First, the collection and processing of input data from users is done through a user-friendly interface where users enter information about themselves. In the second stage, artificial intelligence algorithms analyze the data to generate personalized training programs. The system uses databases of exercises, fitness metrics, and load recommendations to optimize the program for each user.

The effectiveness of the proposed workouts is evaluated in real time by entering user feedback. The AI models and adapts the training plan based on the user's progress, automatically adjusting the intensity and types of exercises to achieve the goals.

Thanks to the integration of artificial intelligence tools, the system not only increases the efficiency of the training process but also makes it more personalized and adaptive, providing users with an individual approach to achieving their fitness goals.

Practical value. The system will provide a high degree of personalization in workout plans, considering various user parameters such as age, weight, goals, and health conditions (injuries/amputations). This allows users to achieve more effective and safe workouts. The system demonstrates the ability to adapt to the user's physical characteristics, which is especially important for individuals with injuries or amputations, ensuring a safe and effective training process.

The use of artificial intelligence for generating workout plans showcases how emerging technologies can improve individualized physical activity planning and adapt to specific user needs. With a personalized approach, users can achieve their fitness goals more effectively, which can positively impact overall health and physical condition.

The system has the potential for further development, including adding new features, integrating with other fitness platforms, or improving artificial intelligence algorithms for even more accurate workout planning.

Conclusions. This work raises an important issue that we face in modern life. This system will help people with various disabilities maintain their physical health, thereby improving their mental health as well. AI tools will be of great benefit as they will allow generating workouts based on user characteristics, taking into account their wishes and adjusting the plan. This area is very important nowadays and there are not many analogous systems that would fully satisfy the needs and desires of users.

1. Su, Z., Ge, S., Li, L., & Su, Y. (2024, May). Review study of integrating AI technology into sports training system. *Educational Administration Theory and Practice*, *30* (5). https://www.researchgate.net/publication/380748116_Review_Study_Of_Integrating_Ai_Technology_Into_Sports_Training_System
2. Owen, R., Owen, J. A., & Evans, S. (2024). Artificial intelligence for sport injury prediction. In *Artificial Intelligence in Sports, Movement, and Health*. Springer. https://www.researchgate.net/publication/379978649_Artificial_Intelligence_for_Sport_Injury_Prediction
3. Biró, A., Cuesta-Vargas, A., & Szilagyi, L. (2024). AI-controlled training method for performance hardening or injury recovery in sports. In *2024 IEEE 22nd World Symposium on Applied Machine Intelligence and Informatics (SAMI)*. https://www.researchgate.net/publication/378226392_AI-controlled_training_method_for_performance_hardening_or_injury_recovery_in_sports
4. Ba, Y., & Liu, Z. (2022). Design and research of physical education platform based on artificial intelligence. *Scientific Programming, 2022* (1), 1–7. https://www.researchgate.net/publication/358792003_Design_and_Research_of_Physical_Education_Platform_Based_on_Artificial_Intelligence

Uliana Pavliuk[1], Volodymyr Yarkun[2], Yuriy Protsyk[3], Oleksandr Storozhuk[4]
[1] *Ukrainian National Forestry University, Lviv, Ukraine, pavlyuk@nltu.edu.ua*
[2] *Ukrainian National Forestry University, Lviv, Ukraine, yarkun@nltu.edu.ua*
[3] *Ukrainian National Forestry University, Lviv, Ukraine, protsyk@nltu.edu.ua*
[4] *Ukrainian National Forestry University, Lviv, Ukraine, storozhuk@nltu.edu.ua*

SMART CAMPUS CONCEPTUAL MODEL FOR MODERN SUSTAINABLE UNIVERSITY

The need for the implementation of the Smart Campus concept for domestic green universities is ripe. In general, smart campus systems can be used to collect and analyze data on students' behavior, academic progress, and other parameters that can help improve the educational process and the quality of education. With the development of IT and IoT, the construction of a Smart Campus as a key component of educational infrastructure is the focus of many studies [2]. Implementation of the Smart Campus idea should be done through application technologies: cloud computing, IoT, AR, and AI to improve the comfort of educational process participants and take into account the goals of sustainable development [1, 2].

Nowadays universities represent the ideal place for the creation of a smart environment. So, the Smart Campus is intended to perform multipurpose uses such as integrated green learning and living environment supported by IoT technologies. The concept of smart university is defined as a small world where sensor-enabled and networked devices work continuously in collaboration to make the infrastructure more efficient [4]. Our main task is to synthesize a Smart Campus conceptual model according to the six criteria of UI GreenMetric World University Rankings and Sustainable Development goals [1]. It will contribute to the systematic collection of data for making strategic decisions for the implementation of sustainable university programs.

The main elements of Smart Campus. Smart Campus involves the use of mobile applications for students, teachers and other campus users that allow interaction with various systems and receive information about campus events, class schedules, available resources and services, and other useful functions. But also the main components of Smart Campus according to sustainable development policy and UI GreenMetric World University Rankings indicator categories (Fig. 1) are infrastructure, energy, waste, water, transportation, and education.

Implementation of this kind of Smart Campus involves the use of IoT technologies. The Smart Campus model includes the use of sensors, data collectors and software to collect, analyze and process data from various locations on campus supported with wireless Internet. This organization of the network makes it possible to implement smart elements in and around housings.

An analysis of existing models [2, 3] of smart campuses was carried out to understand the needs and functional components of typical green university smart campuses and also its certain challenges and limitations. The implementation of the Smart Campus concept requires a comprehensive approach and the use of various technologies and methods to ensure its effective functioning. Therefore, when creating the model, an analysis of various infrastructure elements (buildings, laboratories, educational facilities, recreation areas), technological solutions (IoT networks, security systems, information panels), as well as behavioral aspects (student movement, resource allocation, etc.) was carried out.

The importance of introducing smart technologies into the infrastructural environment of the Smart Campus model has been revealed multiple times, so it will allow the automation of the processes of monitoring and management of various systems, such as lighting, heating, ventilation, security, etc., and will ensure their more efficient use to reduce resource costs.

The introduction of smart technologies into the educational environment has been identified as a key factor in improving the quality of education and streamlining the management of academic processes. Table 1 reflects the process of functioning of the smart campus components from the moment of collecting the necessary information to the stage of forming a decision and obtaining the expected effects.

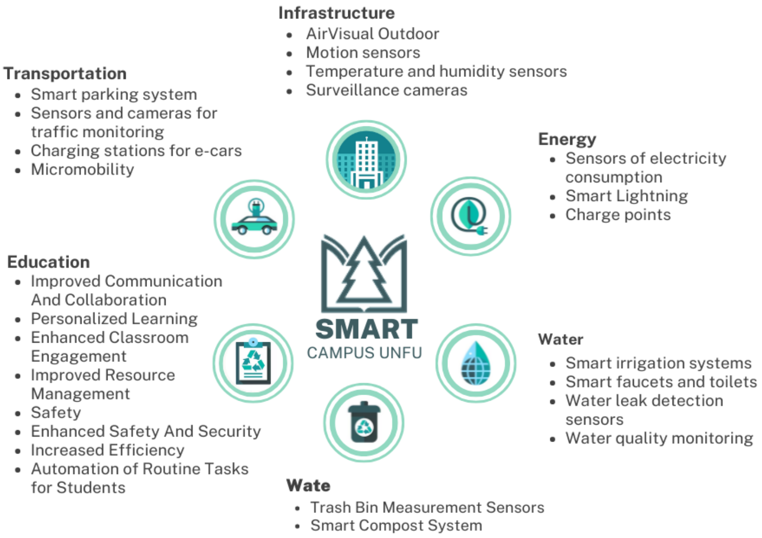


Fig. 1. Components of Smart Campus.

Table 1. Smart campus decision-making processes, collected data, tools and impact of decision-making

Process	Collected data	Data collection tools	Making decisions	Decision effect
Management of energy resources and infrastructure	- Energy consumption of buildings; - Intensity of use of premises; - Climatic parameters; - Equipment condition	- Energy consumption meters; - Presence sensors; - Climate sensors; - Intelligent equipment monitoring systems	- Automatic control of lighting and heating; - Maintenance planning; - Optimization of resource and space allocation	- Reduction of energy consumption costs; - Effective use of premises; - Reduction of wear and tear of equipment
Security and access control	- Entry and exit data; - Video surveillance; - Door/window opening sensors; - Signals from emergency systems	- Access control systems; - Surveillance cameras; - Door/window opening sensors; - Fire alarm system	- Responding to unauthorized access; - Dynamic access control; - Planning of patrols of security services	- Increasing the level of security; - Prevention of unauthorized access; - Reducing the risks of emergency situations
Management of the educational process	- Class schedule info; - Occupancy of classrooms and laboratories; - Level of student activity	- Schedule management systems; - Attendance monitoring systems; - Interactive learning platforms	- Automatic schedule adjustment; - Selection of spaces for classes; - Personalized educational trajectories	- Increasing the effectiveness of training; - Uniform use of auditoriums; - Increasing student engagement
Management of student life and services	- Data on reservation and use of services; - User satisfaction ratings; - Demographic and behavioral data	- Resource booking platforms; - Questionnaires, feedback forms; - CRM systems for processing requests	- Dynamic management of services; - Planning of events and activities; - Personalized recommendations	- Increasing student satisfaction; - Optimization of resource allocation; - Improvement of service efficiency
Environmental monitoring	- Environmental indicators; - Data on the state of vegetation; - Volumes of waste	- Environmental sensors (CO_2 etc); - Water supply monitoring systems; - Waste management systems	- Planning measures to reduce environmental impact; - Optimization of waste utilization and processing	- Reduction of environmental impact; - Improving the quality of the environment; - Environmental sustainability of the campus

Smart Campus Concept Model. The Smart Campus concept model is an innovative solution for creating an educational environment that includes means of collecting, processing and presenting data. The architecture of the Smart Campus is shown in Figure 2 and describes three components:

1. Information collection devices (sensors) that collect various data from the environment, such as temperature, humidity, illumination, air pollution, traffic, and others.
2. A data processing device (master controller) that receives data from sensors and processes them to ensure efficient management of the campus.

3. Data presentation tools (monitors) that display the information collected and processed by the controller.

All these components interact with each other and cooperate to ensure optimal learning conditions and comfort for students and teachers. The data collected by sensors can be used to control heating, ventilation, and air conditioning systems, as well as to monitor indoor air quality. To develop more efficient heating and ventilation systems in modern campus buildings, it is worth considering heat and moisture transfer research [5].

Fig. 2. Smart Campus Concept Model

Conclusion. A conceptual model of Smart Campus has been developed. It will allow collecting, analyzing and processing data from various sources on the campus and, as a result, will ensure the optimal use of resources and increase the efficiency of the campus. The Smart Campus concept has emphasized how the IoT can enhance university management efficiency, improve productivity, and provide a more comfortable and secure learning environment for students and staff corresponding with the goals of sustainable development.

1. Lavnyy, V., Pavliuk, U., & Yarkun, V. (2023). UNFU Sustainable management in the conditions of post-pandemic and war times in Ukraine. IOP Conference Series Earth and Environmental Science, 1194(1), 012032. doi: 10.1088/1755-1315/1194/1/012032
2. Dong, Z. Y., Zhang, Y., Yip, C., Swift, S., & Beswick, K. (2020). Smart campus: definition, framework, technologies, and services. IET Smart Cities, 2(1), 43–54. doi:10.1049/IET-SMC.2019.0072
3. Sari, M. W., Ciptadi, P. W., & Hardyanto, R. H. (2016). Study of smart campus development using Internet of Things Technology. Proceeding of the Electrical Engineering Computer Science and Informatics, 3(1). doi:10.11591/eecsi.v3i1.1126
4. Hossain, I., Das, D. & Rashed, M. G. (2019). Internet of Things Based Model for Smart Campus: Challenges and Limitations. In Proceedings of the 2019 International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering (IC4ME2), Rajshahi, Bangladesh, 1–4.
5. Sokolovskyy, Y., Levkovych, M., Yarkun, V., Protsyk, Y. & Nechepurenko, A. (2020). Parallel Realization of the Problem of Heat and Moisture Transfer in Fractal-structure Materials. 2020 IEEE 15th International Conference on Computer Sciences and Information Technologies (CSIT), 86-90. doi: 10.1109/CSIT49958.2020.9322021.

Mykyta Pliukalov[1], Ihor Prots'ko[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, mykyta.pliukalov.knm.2018@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, ihor.o.protsko@lpnu.ua

PERSONALISED VOICE CONTROL SUBSYSTEM FOR A SMART HOME

Introduction. In modern smart systems, a voice control subsystem is an integral component. With the advancement of the Internet of Things (IoT), voice assistants now play a significant role in managing various devices, from smartwatches to household appliances in smart systems. The implementation of voice control subsystems facilitates voice interaction by covering the management of IT services, the execution of various operations and business functions, device control, and more [1]. Modern voice assistants not only possess a vast database of commands but also adapt their actions over time through user interaction, gradually aligning them with individual preferences. The versatility of the voice control subsystem lies in positioning natural language as a key strategy, extending its utility across various industries.

The ability of the voice control subsystem to analyse and recognize a specific voice opens up new possibilities for personalised interaction. This is particularly important in the context of an interactive assistant used for individual interaction in various fields, including business, education, entertainment, and domestic life. It is worth noting that a personalised approach to voice control in modern smart systems is becoming an essential component of their efficiency.

The aim of this work is to develop the structural diagram of a voice control subsystem and to implement its key parts programmatically, with a focus on personalising the approach to voice control based on the user's voice characteristics, in order to ensure reliable interaction with computers and electronic systems.

Materials and methods. Leading IT companies worldwide pay considerable attention to developing services and APIs for performing various functions via voice assistants, including voice control. Among the notable developments for voice services are: Amazon Alexa, a popular voice assistant available on a range of Amazon devices [2]; Google Assistant, which can perform various tasks, such as answering queries and managing smart household appliances [3]; Apple Siri, which can be used for tasks such as setting alarms and more [4]; and Microsoft Cortana, which can be used for managing smart household appliances and answering queries [5]. These software tools enable integration with their services and interaction with compatible devices to make calls and send messages, supporting multiple users and profiles across a wide range of smart home gadgets. However, these software tools can only be used on devices from specific IT companies, which makes them less universal. There are also concerns about the security and privacy of these systems due to the possibility of eavesdropping and accidental activation. Voice control users also express the need for user-friendly design and well-documented interfaces to facilitate further effective configuration and use of the implemented tools.

The structural diagram of the personalised voice control subsystem, aimed at personalising the approach to voice control, is presented in Fig. 1. The User of the subsystem can also act as an Administrator, performing the configuration of the voice control subsystem.

The data flow begins when the User issues a voice input containing certain commands or requests to be processed by the subsystem. The user's voice input (Voice command) is directed to the speech recognition block (Speech-to-text), which is responsible for converting spoken words into text data. The transcribed text is sent to the command processing mechanism (Command detection), where the text is analysed to understand the user's intent and identify keywords in the command. The subsystem cross-references the identified keywords with the main command table (Command table) stored within the subsystem. The command table contains entries that indicate keywords and corresponding actions that the subsystem should perform. Based on the identified keywords (or their absence), the appropriate action is determined, provided there is a positive confirmation of personalization (Y/N) and the action corresponds to the user's permissions, determined by Voice ID.

User permissions are set by the Administrator during configuration and are checked in the Command table according to the Voice ID.

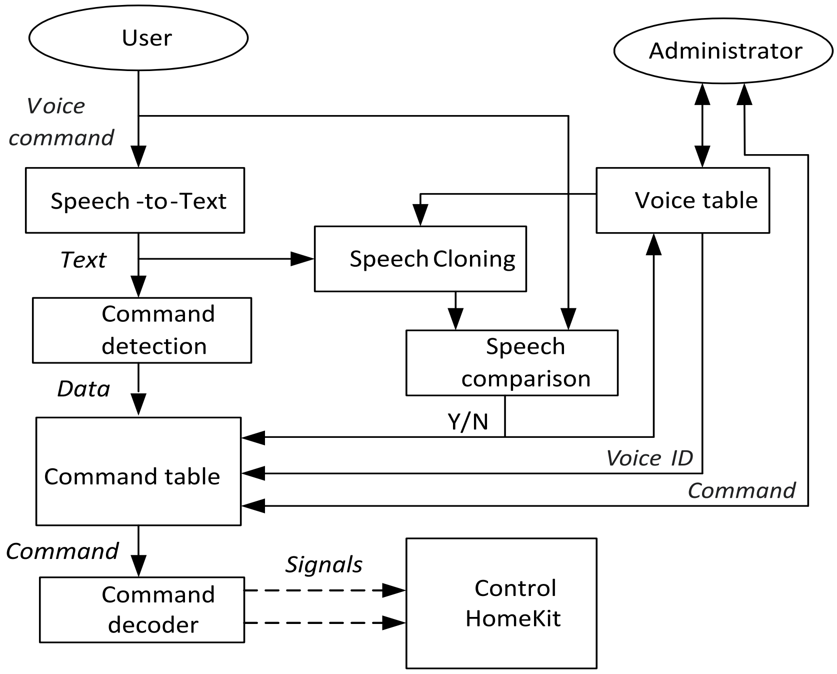


Fig. 1. Structural Diagram of the Personalized Voice Control Subsystem

In the Speech comparison block, the correspondence of the voice command (Voice command) to one of the personalised voices in the voice table (Voice table) is determined. As a result of the match, the corresponding Voice ID of the personalised voice is identified. In the Speech Cloning block, based on the command text and personalised voice samples from the Voice table, a synthetic voice command is generated and sent to the Speech comparison block to be compared with the original voice command (Voice command). If the match is successful, the command code (Command) is selected and decoded in the Command decoder block into the appropriate signals for controlling smart home devices (HomeKit).

The structural diagram of the subsystem ensures that user voice commands are understood, processed, and executed, enabling reliable interaction with electronic systems in the smart home. The command table acts as an important reference point, guiding the subsystem in determining the appropriate responses and actions, ultimately contributing to dynamic and personalised control of the smart home system.

The data flow between the Administrator and the tables (Command and Voice table) includes a series of interactions facilitated by the Administrator via the settings panel. This process requires the Administrator to authorise the management of commands and oversee voice functions within the system. Upon successful authorisation, the Administrator can add, edit, or delete commands in the subsystem. This involves creating or editing the necessary fields to generate a command in the subsystem's command table. The command code (Command), as a unique identifier, is used in the Command decoder block to generate and transmit control signals (Signals).

When adding a new voice, the Administrator not only creates the necessary fields for generating a personalised voice in the voice table (Voice table) but also uploads audio files containing the voice recordings of a specific individual. The system uses these uploaded audio files for subsequent cloning and synthesis of a voice command with unique characteristics. The Voice ID is critical for future actions related to the added unique voice. Based on the audio files from the voice table (Voice table), the process of cloning and synthesising the voice command is initiated. In the Speech Cloning block, a machine learning algorithm is employed, referencing these audio files to create a new voice with characteristics closely matching the provided recordings. The Administrator is given all the necessary tools to control and manage commands and voices within the subsystem.

Results. At the stage of development, the main blocks of the structural diagram (Fig. 1), the libraries are used that support both reading and playback of voice messages in Ukrainian. To develop speech recognition in the interactive voice assistant system with voice cloning capabilities, the Web Speech API [6] library was chosen. This library provides a number of useful functions for voice recognition in interactive voice systems. For example, the onresult() function serves as an event handler and is called when speech recognition is complete. The handler receives recognition results as an array of results.

For generating the voice based on linguistic characteristics, a library written in Python is used. This Python module is designed for cloning a specific person's voice and generating unique cloned voice characteristics. The library is built on the SV2TTS model [7], which consists of separate parts, each trained independently. A speaker encoder and voice synthesiser have been developed. The speaker encoder's task is to take an input sound, encoded as mel-spectrogram frames of a specific speaker, and determine the data, so to speak, embedded speaker data, which are recorded. The synthesiser is also part of the SV2TTS model, which analyses the input text of the voice command to create mel-spectrograms. The synthesiser accepts a sequence of text—this is matched with personalised phonemes and the embedded data elements produced by the speaker encoder. As a result, using the Tacotron 2 architecture [8], frames of the periodic mel-spectrogram of the voice command are synthesised. In the Speech comparison block, the comparison of the synthesised mel-spectrogram and the mel-spectrogram of the original voice command (Voice command) is performed. After matching the voice characteristics, user personalisation takes place, followed by the execution of the voice command.

Conclusions. The developed structural diagram of the voice control subsystem focuses not only on analysing user commands but also on enhancing overall security by creating a more personalised and natural interaction. The developed structure of the voice control subsystem meets the requirement of adaptive functionality to satisfy individual preferences and demands, as well as the requirement for secure personalisation in control systems.

1. Teslyuk, V., Kazarian, A., Kryvinska, N., & Tsmots, I. (2021). Optimal artificial neural network type selection method for usage in smart house systems. Sensors (Switzerland), 21(1), 47, 1–14. https://doi.org/ 10.3390/s21010047
2. Amazon Science. (2022). The engineering behind Alexa's contextual speech recognition. https://www.amazon.science/ latest-news/the-engineering-behind-alexas-contextual-speech-recognition
3. Google Cloud. (2023). Supported voices and languages. https://cloud.google.com/text-to-speech/docs/voices
4. Apple Machine Learning Research. (2017). Deep learning for Siri’s voice: On-device deep mixture density networks for hybrid unit selection synthesis. https://machinelearning.apple.com/research/siri-voices
5. Microsoft Azure. (2022). Text to speech. https://azure.microsoft.com/en-us/products/ai-services/text-to-speech
6. Mozilla Developer Network. (2022). Google Web Speech API. https://developer.mozilla.org/en-US/docs/Web/API/Web_Speech_API/Using_the_ Web_Speech_API
7. Data Science Central. (2021). Voice cloning: Corentin's improvisation on SV2TTS. https://www.datasciencecentral.com/voice-cloning-corentin-s-improvisation-on-sv2tts/
8. Google Research. (2017). Tacotron 2: Generating human-like speech from text. https://blog.research.google/2017/12/tacotron-2-generating-human-like-speech.html

Ihor Prots'ko[1], Roman Rykmas[2], Aleksander Gryshchuk[3]
[1]Lviv Polytechnic National University, Lviv, Ukraine, ihor.o.protsko@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, roman.v.rykmas@lpnu.ua
[3]LtdC "SoftServe", Lviv, Ukraine, ocr@ukr.net

ANALYSIS OF DHT KERNEL STRUCTURE FORMATION FOR SYNTHESIS OF 15-POINT FAST ALGORITHMS

Introduction. The discrete Hartley transform (DHT) was developed to keep the main advantages of fast Fourier transform (FFT), such as performing convolution based on point-wise multiplications [1], but without the additional complexity of working with imaginary numbers. Several methods have been proposed to accelerate the computation of convolution, including the reduction of arithmetic operations via fast Hartley transform (FHT). The domains produced by the Fourier, Hartley, and Cosine transforms have been used to design different deep neural network architectures [2]. The DHT is used to elaborate a residual convolutional neural network, building a faster regression model for the image super-resolution problem, where despite reaching an improvement in the network speed, this work still adopts a simplification of the convolution theorem of the DHT [3].

The DHT computes a linear combination of weighted cosine functions. The direct (1) discrete Hartley transform is written as:

$$H(k) = \sum_{n=0}^{N-1} x(n)\, cas[(nk)2\pi/N], \tag{1}$$

where $H[k]$, $0 \le k \le N$-1, is the Hartley coefficient of index k and $x[n]$, $0 \le n \le N$-1 is the input data of index n of a size N of transform. This type of transform is named after R. Hartley, who in 1942 published an article about a pair of integral transforms, using the function he introduced – $cas(\theta)$. The function $cas(\theta) = cos(\theta) + sin(\theta)$ is a basis function of the transform.

Many fast Hartley transform algorithms were based on already well-known FFTs, which developed in parallel and complemented each other. One approach of efficient algorithms gives the possibility to compute DHT through cyclic convolutions. Special attention is paid to the application of DHT based on cyclic convolutions when creating the VLSI, where microcircuits based on this strategy saved the area of the crystal [4].

Materials and methods. The method of DHT algorithm synthesis based on cyclic convolutions requires the formation of the block structure and its analysis for the determination of the minimum number of cyclic convolutions. This method development on examples analyzing the kernel block structure formation for 15-point DHT, aimed at efficient computation, and a discussion of runtime benchmarking of the DHT_CC program, are presented.

Synthesis of a fast 15-point DHT algorithm, in this case, is important because, in the harmonic analysis of a 60 Hz power signal, it is convenient to assume $N = 15i$, where $i = 2, 3,..$, which solves the problem of sampling a non-integral number of periods and thus spectral leakage. Let us consider, based on the generalized methodology [5], the synthesis of the 15-point DHT algorithm using cyclic convolutions and its performing computation.

Rader first proposed the approach of reducing the kernel of transform to convolution-like structures for calculating the DFT of prime sizes [6]. With the development of fast convolution algorithms [7], this approach for the synthesis of fast transforms has been widely applied to prime or composite values of sizes. By using efficient algorithms for the cyclic structures of the kernel of transform, a fast algorithm for computing the DHT transforms can be obtained.

In the paper [5] the general technique for efficient computation of DHT of sequences of an arbitrary size using cyclic convolutions is considered. The formed hashing cycles in the algorithm define the partitioning of the harmonic kernel into the square submatrices, which can be made through shuffling of the rows and columns. The proposed hashing cycles, used in the algorithms of synthesis, are more versatile and generally better in terms of indexing mapping in comparison with the existing algorithms.

According to the method, the hashing cycles $P(k)$ correspond to the cyclic decomposition of substitution, which is based on the substitutions of rows/columns from the data of arguments of the basis function in the form

$$P(k) = P_1(k)\ P_2(k)\ ...\ P_l(k)\ = (k_{11}, k_{12}, ...k_{1L1})\ (k_{21}, k_{22}, ...k_{2L2})...(k_{k1}, k_{k2}, ...k_{kLt}), \tag{2}$$

where $k = (L1+L2+...+Lt)$ is the size of hashing cycles $P(k)$, Li is the size of a cycle, t is the number of cycles, k_{ij} is an element of a cycle $Pi(k_{ij})$, $i = 1,2,\ldots,l$, $j = 1,2,\ldots,L_j$.

The parameters of hashing cycles $P(k)$ characterize accordingly the modified structure within the transform kernel. For the synthesis of FHT, the structuring of the kernel in the form of a set of cyclic blocks is used by means of the permutation of its rows and columns.

For example, to form a substitution of DHT for size N=15, we select columns with n_1=1 and n_2=2 of the matrix of arguments. The standard form of cyclic decomposition of substitution uses a notation in which the cycles are written in the order of increasing their minimal elements, with each cycle having its smallest element first or vice versa:

$$P(15) = (0)\ (1, 2, 4, 8)\ (5, 10)\ (6, 12, 9, 3)\ (7, 14, 13, 11).$$

The hashing cycles $P(k)$ can be considered as a consecutive set of cycles $Pi(k_{ij})$, $i = 1,2,...,l$. Changing the matrix structure of the kernel of transform in case of changing the sequence of cycles $Pi(k_{ij})$ in (7) will not change the result of the transform, provided that the sequence of input data $x(n)$ is changed accordingly.

Results. The software implementation of the synthesis of DHT algorithms allows you to form various substitutions by rows or columns of the square matrix of integer arguments of the basis functions because according to the theory, there can be more than one primitive element. This will allow an optimization of the structure in the design process, reducing computational complexity and providing high-performance indicators for short sizes of DHT transforms.

Thus, having formed the standard form $Pi(k)$ of the cyclic decomposition of substitution (2), it is necessary to perform an analysis of the block structures of the kernel for the corresponding sequences of $Pi(k)$ cycles and the corresponding cyclic shifts of elements k_{ij} in these cycles. Analysis of block structures of the kernel will allow us to determine the lowest computational complexity of the DHT transform.

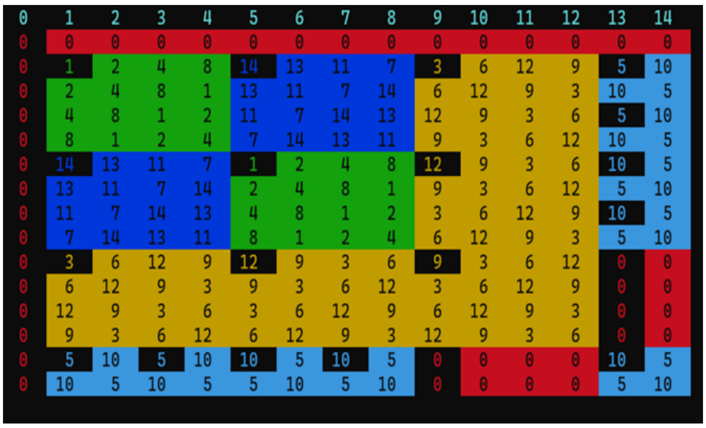
If we reformulate the standard form of cyclic substitution $P_{1,2}(15)$ in the form

$$P_{1,2}(15) = P_0(k)\ P_1(k)\ P_2(k)\ P_3(k)\ P_4(k) = (0)\ (1, 2, 4, 8)(3, 6, 12, 9)(5, 10)(7, 14, 13, 11)$$

with the possibility of changing the sequence of cycles $P_i(k)$ and the cyclic shift of elements k_{ij} in the cycle $P_4(k)$ to the form

$$P'_{1,2}(15) = (0)\ (1, 2, 4, 8)\ (14, 13, 11, 7)\ (3, 6, 12, 9)\ (5, 10).$$

In accordance with obtained hashing cycles $P'_{1,2}(15)$, the structure of the kernel of DHT consists of the cyclic blocks as shown in Fig. 1.



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	2	4	8	14	13	11	7	3	6	12	9	5	10
0	2	4	8	1	13	11	7	14	6	12	9	3	10	5
0	4	8	1	2	11	7	14	13	12	9	3	6	5	10
0	8	1	2	4	7	14	13	11	9	3	6	12	10	5
0	14	13	11	7	1	2	4	8	12	9	3	6	10	5
0	13	11	7	14	2	4	8	1	9	3	6	12	5	10
0	11	7	14	13	4	8	1	2	3	6	12	9	10	5
0	7	14	13	11	8	1	2	4	6	12	9	3	5	10
0	3	6	12	9	12	9	3	6	9	3	6	12	0	0
0	6	12	9	3	9	3	6	12	3	6	12	9	0	0
0	12	9	3	6	3	6	12	9	6	12	9	3	0	0
0	9	3	6	12	6	12	9	3	12	9	3	6	0	0
0	5	10	5	10	10	5	10	5	0	0	0	0	10	5
0	10	5	10	5	5	10	5	10	0	0	0	0	5	10

Fig. 1. The block structure of the matrix of arguments for kernel DHT for size N = 15 is formed by the hashing cycles for $P'_{1,2}(15)$

As a result of the analysis of identical submatrices in the structure of the base matrix (Fig. 1), the size and number of cyclic convolutions for DHT with N=15 are 2-point convolutions: 2 and 4-point convolutions: 5.

However, in the structure of the basic matrix, a block-cyclic arrangement of $P'_{1,2}$ (15) with cycles (1, 2, 4, 8) (14, 13, 11, 7) was formed, which allows us to shorten the computation process using fast 2-point cyclic convolution. Therefore, the presence of a block-cyclic structure in the kernel of the transform (Fig. 1) leads to a reduction in the number of cyclic convolutions in the process of synthesizing the DHT algorithm for N=15. The program codes of algorithms formed by hashing cycles $P_{1,2}$(15) and $P'_{1,2}$(15) are optimized for the set of commands AVX 256 inherent in these computing platforms.

For the rigorous evaluation and comparison of the execution of 15-point DHT algorithms are used the computing platforms Intel®Core™i9-10980XE@3,0GHz with the microarchitecture Cascade Lake-X, Intel®Core™ i9-13900K@3,0GHz with hybrid microarchitecture Raptor Lake, and Apple®Core™ M2 Pro@3,5GHz with the microarchitecture ARM. The performance benchmarking program includes the following computation algorithms: direct DHT_pl using prepared cas() values; DHT_FFTW from the FFTW3.3.10 library using the optimized source plan code; DHT_P1 formed by hashing cycles $P_{1,2}$(15) and DHT_P'1 formed by hashing cycles $P'_{1,2}$(15) both using prepared *cas*() values. The runtime benchmarking of the DHT for randomly selected data of the size N=15 is represented in Table 1 with 10^6 trials. The average times in microseconds of program execution of 15-point DHT are presented on CPU i9-10980XE, i9-13900K, and Apple M2 Pro, respectively.

Table 1. The average time (μs) of computation DHT for N=15

DHT_pl	DHT_FFTW	DHT_P1	DHT_P`1
0.239274355	0.031922170	0.037586585	0.028164896
0.135796261	0.024468613	0.026776981	0.021382713
0.099333403	0.015722319	0.016593792	0.012296162

By Table 1, the program implementation of the synthesized algorithm DHT formed by the hashing cycles $P'_{1,2}$(15) is executed steadily faster than DHT_FFTW for size N=15.

Conclusions. Block structures are often used to optimize computations and efficiently use memory in signal processing. A similar approach to the synthesis of the fast algorithm on the analysis of the formation of a block kernel using the hashing cycles can be used for other sizes of DHT. The specific structures and algorithms used for the DHT can vary, and researchers often explore different approaches to optimize computations or meet specific requirements.

1. Tuan, N. M., & Tuan, P. D. (2009). Generalized convolutions relative to the Hartley transforms with applications. *Scientiae Mathematicae Japonicae*, 70(1), 77-89. https://doi.org/10.32219/isms.70.1_77
2. Reis, E., & Benlamri, R. (2021). Accelerating Convolutional Neural Network using Discrete Orthogonal Transforms. https://doi.org/10.36227/techrxiv.14593686.v1
3. Mozafari, S. H., Clark, J. J., Gross, W. J., & Meyer, B. H. (2021). Implementing Convolutional Neural Networks using Hartley stochastic computing with adaptive rate feature map compression. *IEEE Open Journal of Circuits and Systems*, 2, 805–819. https://doi.org/10.1109/OJCAS.2021.3123899
4. Tsmots, I., Rabyk, V., Kryvinska, N., Yatsymirskyy, M., & Teslyuk, V. (2022). Design of the Processors for Fast Cosine and Sine Fourier Transforms. *Circuits, Systems, and Signal Processing*, 41(9), 4928–4951. https://doi.org/10.1007/s00034-022-02012-8
5. Prots'ko, I. (2014). Algorithm of efficient computation of generalized discrete Hartley transform based on cyclic convolutions. *IET Signal Processing*, 8(4), 301-308.
6. Rader, C. M. (1968). Discrete Fourier transform when the number of data samples is prime. In *Proceedings of the IEEE* (Vol. 56, pp. 1107–1108).

Andrii Pukach[1], Vasyl Teslyuk[2]

[1]*Lviv Polytechnic National University, Lviv, Ukraine, andriipukach@gmail.com*
[2]*Lviv Polytechnic National University, Lviv, Ukraine, vasyl.teslyuk@gmail.com*

MATHEMATICAL MODEL FOR ANALYSING THE INFLUENCE OF IMPACT FACTORS ON THE SOFTWARE COMPLEXES' SUPPORT

Introduction. The main particular scientific and applied problem investigated in this paper is the loss of impact factors boundaries inside the appropriate proposed models of software complexes' support automation as a result of the use of the artificial neural networks (ANN) – namely the multilayer perceptron (MP) ANN type [1]. This loss is caused by the very nature of the MP itself, as according to its nature, all hidden layers' neurons of the MP do not carry any functional-semantic meaning at all, serving only for functional and operational purposes. Thus, the main goal and purpose of the presented paper is to introduce the developed mathematical model, which provides possibilities for analyzing and restoring the impact factors' boundaries, lost as a result of the use of the MP ANN inside the appropriate proposed models of software complexes' support automation. In turn, this particular problem is only a part of a more global scientific and applied problem of software complexes' support automation, which is extremely relevant nowadays with the increasable complexity of various software complexes [2-3]. Automation has already fully encapsulated into a significant number of directions around software complexes' support, among which the following most significant, basic, and key directions can be distinguished, in particular:

- Software testing automation, presented in research [4];
- DevOps automation, represented in works [5–9];
- Decision-making automation, considered in researches [10–11].

Description of input data for the developed mathematical model. The main input data for the developed mathematical model are the ones presented and described below:

1. Existing, developed, and already trained on the general dataset, appropriate MP model representing the interpretation of the subjective perception of the object of software complexes' support. It is also important to note that data for training and testing the MP model must always be prepared and represented in a depersonalized and normalized form – for security reasons when processing any sensitive data (which should always be previously depersonalized), as well as for meeting the requirements for MP ANN incoming data (which should be normalized);
2. Obtained results of testing this MP model on a general dataset, as well as testing results for separate individual (local) datasets for the isolated influence of each individual impact factor (Dimension Factor). MP testing results should contain the following components:
   - *Component 1*: the identifier of the correctness of the obtained MP output result compared to the expected one – since we are interested only in the correct results of the MP model testing, where the actual result coincides with the expected one, while we should simply ignore all incorrect results and do Not consider them into account and do Not include them as input data for our mathematical model (this is extremely an important criterion of the correctness of functioning of the developed mathematical model: incorrect results of MP testing must Not enter our mathematical model in any case, otherwise it will lead to receiving incorrect resulting data at the output of our mathematical model);
   - *Component 2*: reverse chains of maximal weights, formed between the activated neurons (marked with the appropriate impact factors): from the output layer's activated correct neuron of the MP – back through all hidden layers' activated neurons – up to the corresponding input layer's activated neuron.
3. Among all these reverse chains of maximal weights, obtained in the previous step, it is necessary to select only unique ones (Unique Reverse Chain - URC). In addition, since the input layer's neurons (as well as the output layer's neurons) in the obtained reverse chains are not in the focus of interest for us from the point of view of their belonging to the impact factors (and we are only interested in the belonging of the hidden layers' neurons to these impact factors) – we can also

reject them (the output and input layers' neurons) from the resulting unique reverse chains, leaving there only the neurons of the hidden layers.

Development of a mathematical model for analysis of the impact factors' influence on the software complexes' support. According to the main idea of the proposed mathematical model presented in this research, the calculation of impact factors' influence probabilities takes place in several subsequent steps. In the *first step*, the percentage ratio of the frequency of occurrence of each of the URC (within its impact factors) to the total number of all cases (within its impact factors) is calculated. Equation (1) below represents these calculations performed in the first step:

$$FoA\_local[i][j] = count(URC[j]\ in\ F[i])\ /\ count(URC[all]\ in\ F[i]), \qquad (1)$$

where $FoA\_local[i][j]$ – frequency of appearance of the [j]-th URC within the [i]-th impact factor, $count(URC[j]\ in\ F[i])$ – the number of occurrences of the [j]-th URC within the [i]-th impact factor, $count(URC[all]\ in\ F[i])$ – is the total number of occurrences of all URC which exist within the [i]-th impact factor.

In the *second step*, the percentage ratio of the frequency of occurrence of each of the URC (beyond the limits of its impact factors only, but instead – on the general dataset) to the total number of all cases (of this general dataset) is calculated. Equation (2) below represents these calculations performed in the second step:

$$FoA\_global[i][j] = FoA\_local[i][j] * FI[i], \qquad (2)$$

where $FoA\_global[i][j]$ – is the frequency of occurrence of the [j]-th URC of the [i]-th impact factor within the global dataset, $FI[i]$ – is the [i]-th impact factor index.

In turn, the impact factor index is calculated according to the following equation (3):

$$FI[i] = count(F[i]\ in\ General)\ /\ count(all\ in\ General), \qquad (3)$$

where $count(F[i]\ in\ General)$ – is the number of occurrences of the [i]-th impact factor within the general/global dataset, $count(all\ in\ General)$ – is the total number of all cases in this global dataset.

	Probability: [0]..-..[1]			
Disputed neuron	1	2	3	4
HLN[3][1]	0.272	0.7252	-	0.00292
HLN[3][3]	-	-	0.7	0.3
HLN[2][0]	0.1726	-	0.8274	-
HLN[2][1]	0.00316	0.99685	-	-
HLN[2][4]	0.0176	0.97845	-	0.00394
HLN[2][2]	-	-	0.673	0.327
HLN[1][1]	0.00852	0.9915	-	-
HLN[1][4]	-	-	0.67	0.33
HLN[0][3]	0.002	0.28766	0.6974	0.013
HLN[0][1]	0.0067	0.72	0.165	0.1086
Non-disputed neuron				
HLN[3][0]	0	1	0	0
HLN[3][2]	0	0	0	1
HLN[1][0]	0	0	1	0
HLN[1][3]	1	0	0	0
HLN[0][2]	1	0	0	0

Fig. 1. An example of results obtained by a developed mathematical model.

In the *third step*, the actual values of belonging of all the disputed neurons to their respective competitive impact factors are calculated according to the following equation (4):

$$PoDNbtCDF[i][k] = Sum(FoA\_global[i][j](Neuron[k] \in URC[j]))\ /$$
$$/\ Sum(FoA\_global[i][j](Neurons[k] \in URC[all]))\ , \qquad (4)$$

where $PoDNbtCDF[i][k]$ – probability of disputed neuron's [k] belonging to a competitive impact factor [I], $FoA\_global[i][j](Neuron[k] \in URC[j])$ – the frequency of appearance of the [j]-th URC of the [i]-th impact factor within the global dataset (for those $URC[j]$, which includes the disputed neuron *Neuron*[*k*]), *FoA_global*[*i*][*j*](*Neurons*[*k*] Є *URC*[*all*]) – the frequency of appearance of the [*j*]-th URC of the [*i*]-th impact factor within the global dataset (for all URSs of all impact factors *URC*[*all*], which includes the disputed neuron *Neuron*[*k*]).

At the output of the developed mathematical model, we will receive specific numerical values which represent the probabilities of the hidden layers' neurons' (of our MP model) belonging to the relevant impact factors. An example of real obtained results is presented in Figure 1 above.

Conclusions. This article is devoted to the developed mathematical model for analysis of the influence of impact factors on the software complexes' support. The main separate particular scientific and applied problem solved by the developed mathematical model is the problem of restoring the boundaries of impact factors (inside the appropriate proposed models of software complexes' support automation), lost as a result of MP ANN encapsulation inside these models. The results obtained with the help of the developed mathematical model provide an opportunity to investigate further more global scientific and applied problems of software complexes' support automation.

1. Vang-Mata, R. (2020). *Multilayer Perceptrons: Theory and Applications*. Nova Science Publishers, NewYork, 143 p. https://www.google.com.ua/books/edition/Multilayer_Perceptrons/eO-QzQEACAAJ?hl=uk
2. Seniv M. M. (2024). A method of automating the process of calculating reliability indicators of software systems and their components. Ukrainian Journal of Information Tecnology, 6(1), 01-08. https://doi.org/10.23939/ujit2024.01.01.001
3. Kernytskyy, O. B., & Teslyuk, V. M. (2023). The synthesis method for specifications and requirements in the process of it project reengineering. Ukrainian Journal of Information Technology, 5(2), 01–08. https://doi.org/10.23939/ujit2023.02.001
4. Trudova, A., Dolezel, M., & Buchalcevova, A. (2020). Artificial intelligence in software test automation: A systematic literature review. In *Proceedings of the 15th International Conference on Evaluation of Novel Approaches to Software Engineering* (pp. 181–192). https://doi.org/10.5220/0009417801810192
5. Humble, J., & Farley, D. (2010). *Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation* (3rd ed.). Addison Wesley, 463 p. https://books.google.com.ua/books/about/Continuous_Delivery.html?id=9CAxmQEACAAJ
6. Wettinger, J., Andrikopoulos, V., & Leymann, F. (2015). Automated capturing and systematic usage of DevOps knowledge for cloud applications. In *Cloud Engineering (IC2E) IEEE International Conference 2015*, 60–65. https://doi.org/ 10.1109/ic2e.2015.23
7. Salameh, H. (2019). The impact of DevOps automation, controls, and visibility practices on software continuous deployment and delivery. *Proceedings of the 2nd International Conference on Research in Management and Economics*, 22–46. http://www.dpublication.com/wp-content/uploads/2019/09/IME-F793.pdf
8. Sushma, T. N., & Ashok Kumar, A. R. (2020). Automation of software development using devops and its benefits. *International Journal Of Engineering Research & Technology (IJERT)*, *9*(6), 430-432. https://doi.org/10.17577/IJERTV9IS060369
9. Mohanty, L., & Tajammul, M. (2022). DevOps CI Automation Continuous Integration. *International Research Journal of Engineering and Technology (IRJET)*, *9*(3), 1221–1226. https://www.academia.edu/79363743/DevOps_CI_Automation_Continuous_Integration
10. Vadlamudi, S. G., Chakraborti, T., Zhang, Y., & Kambhampati, S. (2016). Proactive Decision Support using Automated Planning. 6 pages. https://arxiv.org/pdf/1606.07841
11. Jaihar, J., Lingayat, N., Vijaybhai, P. S., Venkatesh, G., & Upla, K. (2020). Smart home automation using machine learning algorithms. In *2020 International Conference for Emerging Technology (INCET)*, 1–4. https://doi.org/10.1109/INCET49848.2020.9154007.

Markiian Rakochyi[1], Mariana Seneta[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, Software engineer in Interlogic, mark.rakochyi@gmail.com
[2]Lviv Polytechnic National University, Lviv, Ukraine, mariana.y.seneta@lpnu.ua

SMART-SYSTEMS SECURITY WITH CORS

Introduction. The rapid growth of smart systems has revolutionized how individuals and businesses interact with technology. These interconnected devices rely heavily on web-based APIs, making API security a critical concern in the era of pervasive computing. The subject of the research is the Cross-Origin Resource Sharing (CORS) mechanism. The goal of the research is to enhance performance and security. The novelty is the custom implementation of the Cross-Origin Resource Sharing mechanism that addresses a critical gap in current systems by identifying that preflight requests are not always logged, which can hinder security monitoring. Additionally, our solution optimizes the use of a caching mechanism, thus improving system performance by minimizing latency and server load.

Materials and methods. To mitigate security risks, Cross-Origin Resource Sharing (CORS) has emerged as a critical security mechanism. CORS enables server-side control over which external domains are allowed to access an API. By implementing strict CORS policies, smart systems can effectively prevent unauthorized access to APIs, ensuring that only trusted domains are permitted to interact with sensitive data and devices.

Smart systems often rely on session cookies or tokens for user authentication. If the Access-Control-Allow-Credentials header is set to true, and a wildcard (*) is used for Access-Control-Allow-Origin, it creates a severe vulnerability.

The CORS protocol involves the browser, the client, and the server, and functions as a layer of control that allows the server to dictate how resources can be shared across domains. Here's a breakdown of how CORS operates:

- Cross-Origin Request: When a browser sends a request to a resource on a different domain (origin), the request is marked as a cross-origin request.
- CORS Headers: The server receiving the request inspects it and, if it wants to allow cross-origin access, it responds with specific CORS headers indicating which domains are allowed to access the resource. These headers might include:
  - Access-Control-Allow-Origin: Specifies which domain(s) can access the resource.
  - Access-Control-Allow-Methods: Specifies the allowed HTTP methods.
  - Access-Control-Allow-Headers: Lists the headers allowed in the request.
  - Access-Control-Allow-Credentials: Indicates whether the request can include credentials
- Browser Enforces CORS: If the server responds with the appropriate CORS headers, the browser allows the cross-origin request to proceed. If the headers are missing or incorrect, the browser blocks the request, protecting the user from unauthorized access.

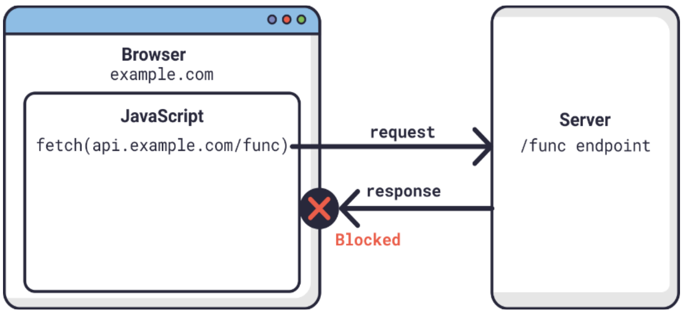


Fig 1. Graphical representation example of CORS mechanism

The results of research. During the investigation, it was found that smart systems often interact with a variety of external origins, which results in preflight requests being frequently triggered. However, it became evident that these preflight requests, despite their importance in controlling cross-origin access, were not consistently included in logging mechanisms. This oversight limits the ability to track and analyze request patterns, leaving potential security risks undetected.

Recognizing this gap, the research emphasized the need for custom logging preflight requests. By capturing data such as request origins, methods, and headers, system administrators can monitor patterns and detect anomalies, such as sudden spikes in requests from a specific origin that may indicate malicious activity. Establishing robust logging practices allows organizations to better understand cross-origin interactions in smart systems and respond proactively to potential threats.

Another important discovery in managing preflight requests is the role of the Access-Control-Max-Age header, which controls how long preflight responses can be cached by the browser. In large smart systems, extending this value can significantly reduce the number of preflight requests, boosting system responsiveness and improving the user experience. However, the investigation revealed that longer caching periods, while beneficial for performance, could pose security risks if outdated permissions are applied after a policy change.

To mitigate this risk, the research suggests implementing a mechanism to reset the max-age value whenever security configurations change. This ensures that clients are prompted to revalidate permissions in a timely manner, keeping security policies current without compromising performance. A well-structured strategy for managing max-age balances operational efficiency with security, ensuring that policy updates are recognized immediately by the system.

Practical value. This research is particularly important for large-scale smart systems where external domains frequently change. By optimizing the handling of preflight OPTIONS requests through efficient caching, the solution improves performance and minimizes server load. Additionally, the custom logging mechanism enhances security by monitoring and analyzing preflight request patterns, enabling the detection of anomalies and potential threats.

Conclusions. As smart systems become increasingly interconnected, ensuring secure communication between different origins is essential to protecting sensitive data and preventing unauthorized control of devices. My research highlights the urgent need for improved logging practices and careful management of caching in CORS implementations. By emphasizing the importance of the handling of preflight requests, organizations can significantly enhance the security and performance of smart systems.

1. Wenz, C. (2022). *ASP.NET security*. *Vol. 4*, *Iss. 4*, 170-181.
2. Kohei, A., Kodai, N., & Mariko, O. (2024). Method resource sharing in on-premises environment based on cross-origin resource sharing and its application for safety-first constructions. *International Journal of Advanced Computer Science and Applications, 15*(5), 1–6.
3. Gedam, M. N., & Meshram, B. B. (2023). Proposed secure hypertext model in web engineering. *Department of Computer Engineering*, 3–18. https://journals.riverpublishers.com/index.php/JWE/article/view/21591/18847.

Roman Rykmas[1], Ihor Prots'ko[2]
[1]*Lviv Polytechnic National University, Lviv, Ukraine, roman.v.rykmas@lpnu.ua*
[2]*Lviv Polytechnic National University, Lviv, Ukraine, ihor.o.protsko@lpnu.ua*

USE OF DISCRETE TRANSFORMS IN CONVOLUTIONAL NEURAL NETWORKS

Introduction. Convolutional neural networks have found wide application in the field of computer vision, in areas such as image classification [1–2], object detection [3], and clustering [4]. At the same time, the complexity of the problems faced by convolutional neural networks is constantly increasing, which in turn requires the creation of networks with a greater number of layers and neurons in them. An increase in these network parameters leads to an increase in computational operations, in particular, convolution. Increasing network complexity and computational costs are a common problem for all proposed convolutional neural network architectures. Training such models requires significant computing resources and time [5].

To speed up the learning of convolutional neural networks, two directions are distinguished: reducing the number of parameters, and increasing the efficiency of computations. Reducing the number of parameters leads to the fact that less memory is used, as well as better generalization of the learning results because it is assumed that a network with a smaller number of parameters does not lose accuracy [6]. An increase in the efficiency of calculations occurs due to the optimal selection of data structures and algorithms; this method of acceleration allows us to achieve maximum results with the available hardware.

The paper considers speeding up the operation of a convolutional neural network's operation based on the convolution operation's application. Various fast computational algorithms are used to calculate the convolution. Convolution plays a key role in digital signal processing. Many efficient algorithms and tools have been developed for fast computations of convolutions and discrete transforms. It is known that the convolution can be computed more quickly using fast Fourier transforms (FFT). In addition to using FFT, the convolution can be computed through other discrete transforms, which can be faster such as discrete cosine transform (DCT) [7].

Materials and methods. Discrete trigonometric transforms are variants of the discrete Fourier transform (DFT), which include the discrete cosine transform (DCT) and the discrete sine transform (DST), the name of which indicates which trigonometric function each of them uses. Their feature is that they represent a discrete signal sequence in the form of a linear combination of trigonometric functions. This allows different parts of the signal to be separated and the energy to be concentrated in the areas where the signal is of great importance, making it efficient for further compression and coding.

DCT is most often used. There are 8 types of DCT, but only the first four are used more often. In practice, DCT-II is widely used, if it is written about the use of DCT without specifying the type, then most likely DCT-II is meant.

DCT-II is calculated according to the following formula:

$$X_k = \sqrt{\frac{2}{N}} C_k \sum_{n=0}^{N-1} \quad x_n \cos\cos\left(\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right), \tag{1}$$

where C_k – is a coefficient equal to $\frac{1}{\sqrt{2}}$ for k = 0 and N or 1 for all other values of k.

There are fast computation methods for computing both the DCT and the DFT. Fast computation of DCT can be organized with the help of models based on block-cyclic structures [7].

DCT is a symmetric transform and can process so-called symmetrically extended sequences, which are denoted as 2N-point sequences, which can be described by the following formulas:

$$\hat{x}_{2N}(n) = x_{2N}(n) + x_{2N}(-n-1) \tag{2}$$

where

$$x_{2N}(n) = \{x_N(n),\ 0 \le n \le N-1 \quad 0,\ N \le n \le 2N-1 \tag{3}$$

Let $x(n)$ and $h(n)$ be a sequence of length M and a filter of length L, respectively, where $M > L$. The corresponding formula for computing a cyclic convolution over an extended $2N$-point sequence can be presented as follows:

$$\hat{x}_{2N}(n) \otimes \hat{h}_{2N}(n) = \frac{1}{N}\sum_{k=0}^{N} \quad C_k^2 X_c(k) H_c(k) \cos \cos \left(\frac{\pi n k}{N}\right), \tag{4}$$

where $X_c(k)$, $H_c(k)$ are discrete cosine transforms $x(n)$ and $h(n)$.

Hence, the first N points of the cyclic convolution of the $2N$-point extended sequences are computed using the N-points of the DCT.

Important is the fact that cyclic convolution has a relationship with linear convolutions, which is striking in symmetry. The cyclic convolution between $\hat{x}_{2N}(n)$ and $\hat{h}_{2N}(n)$, which according to (2) are extensions of $x_N(n)$ and $h_N(n)$, is expressed by the superposition of four linear convolutions as:

$$\hat{x}_{2N}(n) \otimes \hat{h}_{2N}(n) = y_{2N}^{(1)}(n) + y_{2N}^{(2)}(n) + y_{2N}^{(3)}(n) + y_{2N}^{(4)}(n) \tag{5}$$

where

$$y_{2N}^{(1)}(n) = \{x_M(n) \otimes h_L(n),\ R_1\ 0, \quad otherwise \tag{6}$$

$$y_{2N}^{(2)}(n) = \{x_M(-n-1) \otimes h_L(n),\ R_2\ 0, \quad otherwise \tag{7}$$

$$y_{2N}^{(3)}(n) = \{x_M(n) \otimes h_L(-n-1),\ R_3\ 0, \quad otherwise \tag{8}$$

$$y_{2N}^{(4)}(n) = \{x_M(-n-1) \otimes h_L(-n-1),\ R_4\ 0, \quad otherwise \tag{9}$$

for $R_i = l_i \leq n \leq l_i + (L + M - 1), i = 1, 2, 3, 4.$

Thus, there are effective methods of computing DCT, which in turn allow efficient computation of cyclic convolutions, which are expressed in 4 symmetric linear convolutions.

DCT-based convolution can be used in the architecture of convolutional neural networks (Fig.1) as follows: the input sequence and the filter are padded with zeros so that operations are performed on $N \times N$ point data sets. Next, the input image and the filter supplemented with zeros are transformed with the help of DCT, after which they are multiplied by elements, and after applying the inverse DCT, we get the computed convolution. The fact that we obtain convolutions (6), (7), (8), (9) as a result of performing a cyclic convolution operation on extended $2N$-point sequences allows us to obtain a feature map for a normal image and a filter, a horizontally inverted image and filter, a vertically inverted image and filter, horizontally and vertically rotated image and filter.

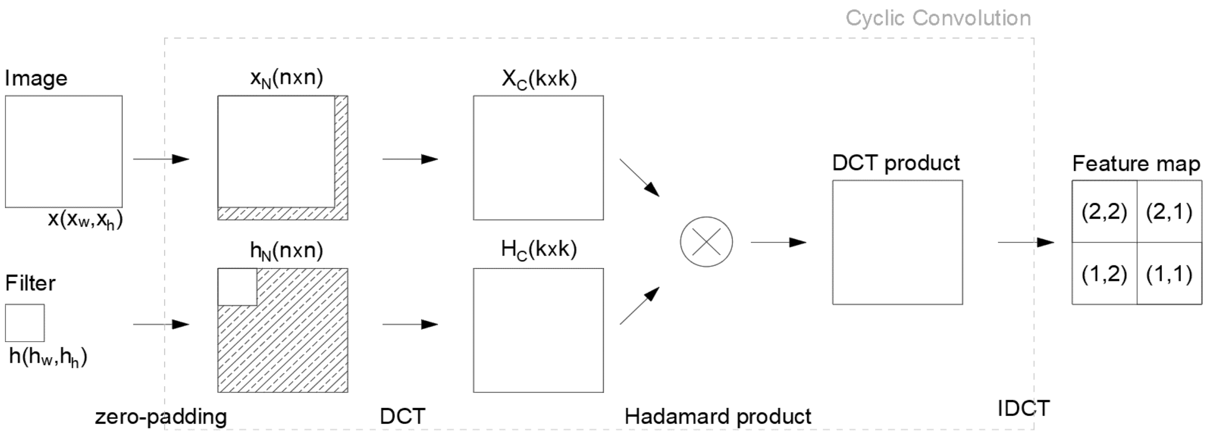


Fig. 1. DCT-based convolution in the convolutional neural networks

Results. During the experimental part of the study, convolution computations were implemented in various ways using DCT-II and DCT-I, and their speed was investigated. In addition to the above convolution computation method, other methods of computation using DCT, and other discrete transforms were also used [8]. In the study the convolution computation was implemented through DFT, DHT, and the DCT in two ways (Fig. 2). A convolution was calculated with a kernel size of 127, 255, 511, 1023. The time of execution of the convolution operation is indicated in seconds.

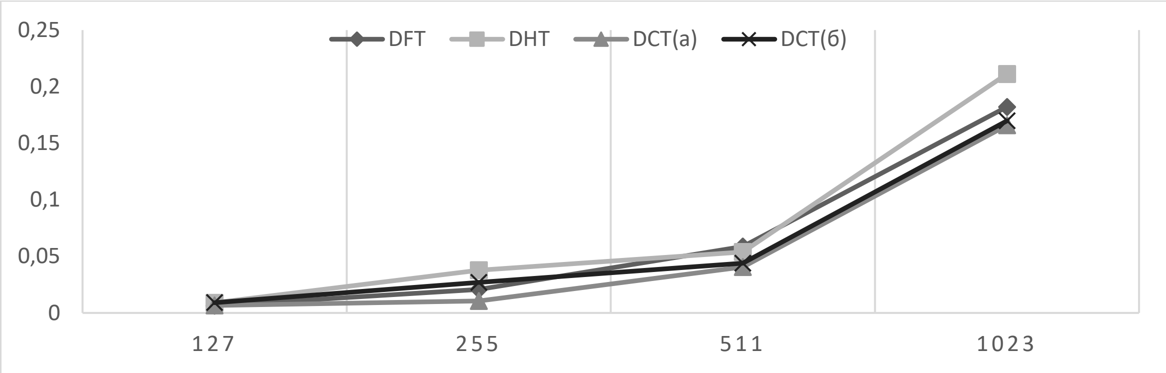


Fig. 2. The time (second) of execution of the convolution through DFT, DHT, DCT for size of 127, 255, 511, 1023

After analyzing the operations that are performed in each case, we can come to the conclusion that the above method will be the fastest since it uses two transforms times. Other methods require more operations to compute the convolution. To further speed up the computation of the convolution, it is suggested to use generating arrays based on block-cyclic structures [9]. They can be used both for DCT and for calculating the factor $cos\ cos\ \left(\frac{\pi n k}{N}\right)$, which is present in formula (4).

Conclusions. Most of the scientific works on accelerating neural network learning are at the research stage, therefore, the search for a compatible effective application of discrete transforms and deep neural networks can provide researchers with new methods and means of accelerating the operation of networks and efficient use of computers.

1. Chen, L., Li, S., Bai, Q., Yang, J., Jiang, S., & Miao, Y. (2021). Review of image classification algorithms based on convolutional neural networks. *Remote Sensing, 13*(22), 4712. https://doi.org/10.3390/rs13224712
2. Bakhshi, A., Chalup, S., & Noman, N. (2020). Fast evolution of CNN architecture for image classification. In H. Iba & N. Noman (Eds.), *Deep neural evolution* (pp. 177–194). Springer. https://doi.org/10.1007/978-981-15-3685-4_8
3. Shah, S., & Tembhurne, J. (2023). Object detection using convolutional neural networks and transformer-based models: A review. *Journal of Electrical Systems and Information Technology, 10*(54). https://doi.org/10.1186/s43067-023-00123-z
4. Huang, H., Wang, C., Wei, X., & Zhou, Y. (2024). Deep image clustering: A survey. *Neurocomputing, 599*, 128101. https://doi.org/10.1016/j.neucom.2024.128101
5. Zhu, H., Akrout, M., & Zheng, B. (2018). Benchmarking and analyzing deep neural network training. *2018 IEEE International Symposium on Workload Characterization (IISWC)*. https://doi.org/10.1109/IISWC.2018.8573476
6. Han, S., Mao, H., & Dally, W. J. (2016). Deep compression: Compressing deep neural network with pruning, trained quantization and Huffman coding. *International Conference on Learning Representations (ICLR) 2016.* https://doi.org/10.48550/arXiv.1510.00149
7. Prots'ko, I. (2013). Algorithm of efficient computation of DCT I-IV using cyclic convolutions. *International Journal of Circuits, Systems and Signal Processing, 7*(1), 1–9
8. Usakova, A., Kotuliakova, J., & Zajac, M. (2002). Using of discrete orthogonal transforms for convolution. *Journal of Electrical Engineering, 53*(9–10), 285–288.
9. Prots'ko, I., & Rykmas, R. (2023). The analysis of models of the block-cyclic structures of the DCT-II core for the synthesis of fast algorithms. In *Materials of the XI International Scientific Conference "Information-Management Systems and Technologies"*, 21–23 September, 2023.

Dmytro Savchuk[1], Anastasiya Doroshenko[2]
[1] *Lviv Polytechnic National University, Lviv, Ukraine, dmytro.y.savchuk@lpnu.ua*
[2] *Lviv Polytechnic National University, Lviv, Ukraine, anaastasiia.v.doroshenko@lpnu.ua*

TRAFFIC ACCIDENTS PREDICTION IN SMART CITIES

Introduction. With the development of technology, smart cities are becoming a reality, providing new opportunities to improve road safety. One of the important tasks that can be solved with the help of intelligent systems is the prediction of road traffic accidents (RTAs). Using big data, artificial intelligence (AI), and analytical tools, we can create models that predict possible accidents, allowing preventive measures to be taken in advance.

Such solutions have the potential to significantly reduce the number of road accidents, minimize medical costs, and improve the overall efficiency of the urban transportation system. This thesis discusses modern approaches to traffic accident forecasting, including machine learning methods, data analytics, and factors that influence the occurrence of accidents.

Problem statement. The object of research is the intellectual transportation systems of smart cities, particularly their ability to analyze and predict road traffic accidents (RTAs). In today's context of growing urbanization and the increasing number of vehicles, road safety is becoming increasingly important, making it necessary to find new solutions to reduce the number of road accidents.

The subject of the study is methods and technologies for predicting road accidents in smart cities using big data, data mining, artificial intelligence (AI), machine learning, and other analytical approaches. The study of factors that affect the likelihood of accidents, as well as tools for their prediction, is a key aspect of this research.

The purpose of the research is to develop an effective intellectual model for predicting accidents in smart cities, which will reduce the number of accidents by preventing risks in advance. To achieve this goal, it is planned to apply modern methods of data analysis, use integrated traffic monitoring systems, and identify key factors affecting road safety.

A dataset consisting of 45 features and about 8 million incidents was used [1] to predict traffic accidents. These characteristics include the time of the event, coordinates, distance of the road incident, city, region, zip code, time zone, temperature, airport, wind, humidity, pressure, visibility, precipitation, weather conditions, amenity, bump, junction, crossing, railway, roundabout, station, stop, period of day, and others.

Main part. In this work, several diverse tasks were solved that covered all stages of working with the dataset. First, a detailed analysis of the dataset was conducted, including a review of its content, identification of the main quantitative and qualitative characteristics, and study of possible types of these characteristics. After that, statistical information about the data was collected, formatted, and checked for zero values. In cases where the amount of missing data exceeded 40%, a mechanism for generating missing data was implemented [2].

Particular attention was paid to studying the number of different types of incidents depending on several factors, such as region of the country, city, year, month, day of the week, and weather conditions. The distribution of incidents by time of day, weather conditions, and duration of events was analyzed in detail. In addition, a correlation matrix was constructed to examine the relationships between the quantitative and qualitative characteristics of the dataset. Qualitative characteristics were converted into quantitative ones by encoding them.

The next step was to split the dataset into training and test samples in the ratio of 70% to 30%, respectively. Next, various machine learning models were researched and developed that could be used to predict the probability of traffic incidents. The models considered included Random Forest, Extreme Gradient Boosting, Gradient Boosting, Logistic Regression, Extra Tree, Decision Tree, MLP Classifier, and others [3].

The modeling results (Table 1) showed that Extreme Gradient Boosting - 85%, Light Gradient Boosting - 84%, and Random Forest - 82% demonstrated the highest accuracy. Decision Tree - 76% and SVM – 73% had the lowest accuracy [4].

After analyzing the results, it was determined that several methods provide approximately the same level of classification accuracy. In such cases, it is advisable to choose a model that is more understandable and explainable. For this study, the Gradient Boosting method was the best choice, as it combines high accuracy with better model interpretation [5].

Table 1. Modeling results

Model	Accuracy	AUC	Recall	Precision	F1	TT
Extreme Gradient Boosting	0.844	0.873	0.415	0.829	0.823	1.950
Light Gradient Boosting	0.839	0.865	0.390	0.823	0.815	0.594
Gradient Boosting Classifier	0.829	0.841	0.356	0.812	0.792	3.871
Random Forest Classifier	0.823	0.824	0.313	0.806	0.777	0.439
Extra Trees Classifier	0.808	0.744	0.299	0.766	0.756	0.539
Logistic Regression	0.797	0.717	0.270	0.728	0.733	2.198
Linear Discriminant	0.795	0.693	0.261	0.720	0.722	0.149
Decision Tree Classifier	0.758	0.652	0.410	0.762	0.760	0.087
SVM	0.733	0.000	0.295	0.719	0.699	0.525

Conclusions. This work demonstrated the potential for predicting traffic accidents in smart cities using big data models and machine learning, with Gradient Boosting proving to be the most effective approach due to its high accuracy (84%) and interpretability. By analyzing a dataset of 45 characteristics and approximately 8 million incidents, the study identified key factors that influence road accidents, such as time, place, and weather conditions, and emphasized the importance of data preprocessing to ensure reliable results [6]. The study showed that predictive models can significantly improve road safety in cities by enabling proactive measures such as adjusting traffic signals and warning of high-risk conditions. Although the study showed promising results, future work should focus on improving the models with real-time data and incorporating additional sources, such as IoT-based traffic monitoring systems, to improve the accuracy of prediction and generalization across different smart cities. Overall, the research contributes to the development of safer and more efficient urban transportation systems.

Acknowledgment. This work was realized within the framework of the Erasmus+ Jean Monnet Chair 2022 «101085612 – DataProEU - Data Protection in the EU».

1. US Accidents (2016 - 2023). (2023). Kaggle: Your Machine Learning and Data Science Community. https://www.kaggle.com/datasets/sobhanmoosavi/us-accidents
2. Doroshenko, A. (2019). Application of global optimization methods to increase the accuracy of classification in the data mining tasks. Computer Modeling and Intelligent Systems, 2353, 98–109. https://doi.org/10.32782/cmis/2353-8
3. Savchuk, D., & Doroshenko, A. (2021). Investigation of Machine Learning Classification Methods Effectiveness. 2021 IEEE 16th International Conference on Computer Sciences and Information Technologies (CSIT), 33-37. https://doi.org/10.1109/csit52700.2021.9648582
4. Boukerche, A., & Wang, J. (2020). Machine learning-based traffic prediction models for Intelligent Transportation Systems. Computer Networks, 181, 107530. https://doi.org/10.1016/j.comnet.2020.107530
5. Alshamrani, F. H., Syed, H. F., & Elhussein, M. A. (2019). Machine learning based model for traffic prediction in Smart Cities. 2nd Smart Cities Symposium (SCS 2019). https://doi.org/10.1049/cp.2019.0195
6. Nagy, A. M., & Simon, V. (2018). Survey on traffic prediction in Smart Cities. Pervasive and Mobile Computing, 50, 148–163. https://doi.org/10.1016/j.pmcj.2018.07.004.

Zoreslava Shpak[1], Mykhailo Shuvar[2]
[1]*Lviv Polytechnic National University, Lviv, Ukraine, zoreslava.y.shpak@lpnu.ua*
[2]*Lviv Polytechnic National University, Lviv, Ukraine, mykhailo.shuvar.oi.2023@lpnu.ua*

SMART HOME NOTIFICATION SYSTEM USING THE TELEGRAM MESSENGER

Introduction. Smart homes, which allow for the efficient use of various home life support systems, are becoming more and more in demand nowadays [1]. For the convenience of users, various remote control tools for smart homes are being developed, mostly via the Internet. Special applications, websites, or other platforms are usually created to allow remote smart home control and management from a phone or computer [2]. However, the implementation of this approach mostly requires communication servers, which can significantly affect the total cost of the system and become a problem for budget projects.

Solutions and results. The solution to this problem can be to use existing platforms. Among them is Telegram, one of the most popular messengers in the world. Telegram messenger provides users with a convenient opportunity to create and use chatbots based on a modern multifunctional API. Such chatbots can be used not only to send and receive notifications but even to create full-fledged control panels. The advantage of creating a bot using this messenger is that Telegram, in addition to working with regular text messages, provides a wide range of functions, such as sending commands, creating built-in notification buttons, specialized keyboards, and others. This functionality makes it possible to develop a user-friendly interface. Another positive aspect is that a big number of potential smart home owners have experience in the application of this messenger, so they don't need to spend time learning a new application.

The general scheme for processing information messages of the smart home system using the Telegram messenger is presented in Fig. 1.

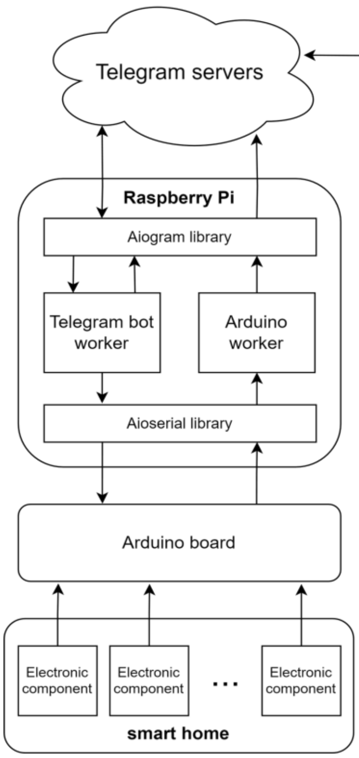


Fig. 1. General scheme of the message transmission system

The lowest level contains components for direct control and interaction with smart home executive mechanisms and devices, sensors, controllers, and other elements. All these components are connected to the inputs or outputs of the Arduino board. The Arduino microcontroller is connected via a USB channel to the Raspberry Pi minicomputer, which allows data exchange via the Serial protocol. An ordinary home computer or laptop, which supports the processing of Python program codes and provides access to the Internet, can be used instead of Raspberry Pi.

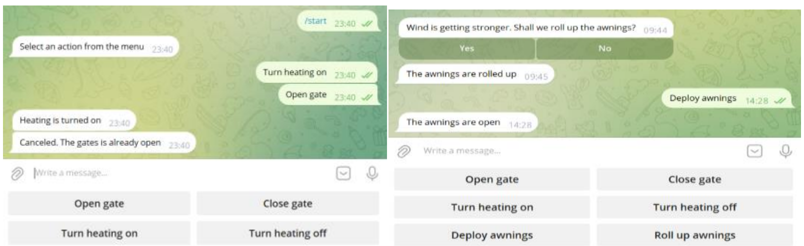
The Arduino board is programmed to control all connected components of the smart home and to exchange relevant data with the computer. In turn, the computer directly interacts with the Telegram bot. It sends messages to the user, which are displayed on the mobile device, and also receives commands from the user, which, after processing, are transmitted to the Arduino board.

Programmatic control over the exchange of messages and commands is implemented using Python. The Aiogram library is used to organize communication with Telegram (transmitting information from the computer and receiving it back from the bot). Data exchange between the computer and the Arduino microcontroller is carried out via the Serial protocol using the Aioserial library.

The main difficulty of the software solution was the need to ensure simultaneous processing of data coming from both Telegram and Arduino. For this purpose, two software handlers were created. The first one, the Telegram bot worker, is responsible for organizing the transfer of data coming from the Telegram messenger to the Arduino board, and the second one, the Arduino worker, manages the data transferred from the Arduino to Telegram. Both handlers run on the same processor core, which requires coordination of their interaction. The synchronous organization of the handlers' work, when requests are processed in turn, leads to inefficient use of the processor's time, which will be idle while waiting for a response from the Telegram server.

Therefore, it was decided to use an asynchronous organization of handlers interaction, which is especially effective when you need to cooperate with the server. At each moment (quantum) of time, the processor works with one of the handlers. If a request to another worker is received at this time, it is added to the queue. As soon as the data from the first handler is sent to the server, the processor switches to work with the new request. It is possible that messages from both Telegram and Arduino will be received simultaneously. In this case, the request whose handler is currently active will be executed first.

Fig. 2 shows two screenshots that demonstrate examples of user communication with the smart home system. Fig. 2a illustrates the case when a user sends certain instructions via the Telegram messenger. These messages are sent to the Telegram worker, which uses the Aioserial library to send the corresponding instructions to the Arduino board. Next Arduino activates the necessary smart home executive mechanisms and through the Arduino worker, reports to the Raspberry Pi about the successful or unsuccessful completion of the task. Then, the processor sends a corresponding notification to the user using the Aiogram library. Another case, when the initiator of communication is the Arduino board, is shown in Fig. 2b. The message from the smart home system about strong winds is complemented by built-in buttons for choosing the answer option.



(a) (b)

Fig. 2. Examples of user communication with the smart home system via the Telegram bot:
a) the user sends two simultaneous instructions;
b) the user gets a warning notification with built-in buttons for selecting an answer

Conclusion. In general, we can conclude that it is not necessary to use a centralized server to organize the exchange of notifications between the user and the smart home system. The task of interaction and remote control can be successfully implemented by using messengers, such as Telegram. The use of chatbots makes the notification system efficient and user-friendly.

1. Number of users of smart homes worldwide from 2019 to 2028. *Statista*. https://www.statista.com/forecasts/887613/number-of-smart-homes-in-the-smart-home-market-in-the-world
2. Mendes, T. D. P., Godina, R., Rodrigues, E. M. G., Matias, J. C. O., & Catalão, J. P. S. (2015). Smart home communication technologies and applications: Wireless protocol assessment for home area network resources. *Energies*, *8*(7), 7279-7311. https://doi.org/10.3390/en8077279.

Ivan Shpakovych[1], **Volodymyr Mokryi**[2]
[1]*Professional College of Information Technologies of the Lviv Polytechnic National University, Lviv, Ukraine, ivan.m.shpakovych@lpnu.ua*
[2]*Lviv Polytechnic National University, Lviv, Ukraine, volodymyr.i.mokriy@lpnu.ua*

INFORMATION TECHNOLOGIES OF WAR-DAMAGED KAKHOVKA RESERVOIR SPACE MONITORING

Introduction. Monitoring the Kakhovka Reservoir, which the Russian occupiers destroyed, is necessary because information must be provided to make management decisions regarding environmental safety and strategies for restoring the reservoir's infrastructure.

The explosion of the Kakhovka HPP by the Russian occupiers resulted in a large-scale man-made disaster, the consequences of which were catastrophic for the southern regions of Ukraine and the southwestern part of the Black Sea. Over 16,000 individuals and 80 settlements were affected [1]. The destruction of the dam resulted in significant human casualties and a large-scale environmental disaster and constitutes a war crime and ecocide. The choice of options for the restoration of the Kakhovka Reservoir is based on comprehensive studies of ecosystem transformations caused by the destruction of the hydroelectric dam, operational monitoring of the further course of this disaster, and environmental and economic forecasting of possible consequences.

Materials and methods. The particularity of utilizing space imagery to obtain new information lies in its visibility and the capacity to study the Earth's surface at varying levels of generalization. Sufficiently substantiated data on the condition of water bodies, in particular large reservoirs, enables the enhancement of management decisions to control and improve the environmental condition of reservoirs. To achieve this, it is essential to have reliable, timely, and comprehensive data on the main morphometric characteristics, ecological parameters, and the state of the water body under study. Most data can be obtained using space-based monitoring materials, including hydrological conditions, dewatered and flooded areas, and residual water bodies.

The current state of the dewatered bed of the Kakhovka Reservoir was investigated using a geographic information system (GIS) analysis of satellite images. The Normalized Difference Water Index (NDWI), obtained from remote sensing data of the Landsat satellite constellation, was used to assess the state of the hydrography of the Kakhovka Reservoir and the state of the dam of the Zaporizhzhya NPP cooling pond. The NDWI is employed to monitor alterations in the water content of reservoirs by utilizing green and near-infrared wavelengths. Index values are generated from satellite images exhibiting maximum reflectivity of water bodies in the green wavelength spectrum and minimum reflectivity in the near-infrared spectrum, where vegetation and soil exhibit maximum values. The presence of a water mirror results in low values of reflection coefficients in the near-infrared and high values in the visible green spectrum. The ratio of these indicators allows for the clear separation of water from other natural objects. The described features of the zonal distribution of light reflection spectra on the water surface areas determine the effectiveness of using multispectral satellite imagery for mapping water bodies. For the visualization of the NDWI index, the technologies of the cross-platform geographic information system QGIS (Quantum GIS) [2] were employed.

The NDWI index was employed to create ecological and cartographic models of the hydrography of the Kakhovka Reservoir. Multi-temporal satellite images with a minimum percentage of cloud cover, obtained from Landsat 8 OLI and Landsat 9 OLI satellites, hosted by the US Geological Survey [3], were used.

Results. The water level of the Kakhovka Reservoir on the hydrographic map synthesized based on the satellite image of 23 August 2020 is taken as a reference. The identified satellite image from 13 July 2023 shows the old Dnipro riverbed and a significant number of shallow water bodies that were left after the water level rose. The structural integrity of the dam of the Zaporizhzhia NPP cooling pond remains unchanged, and the water level in the cooling pond has not dropped significantly in comparison to the control visualization. A hydrographic map of ecosystem transformations due to the drainage of the Kakhovka reservoir bed was generated from satellite images of 2 October 2023 using the NDWI index (Fig. 1).

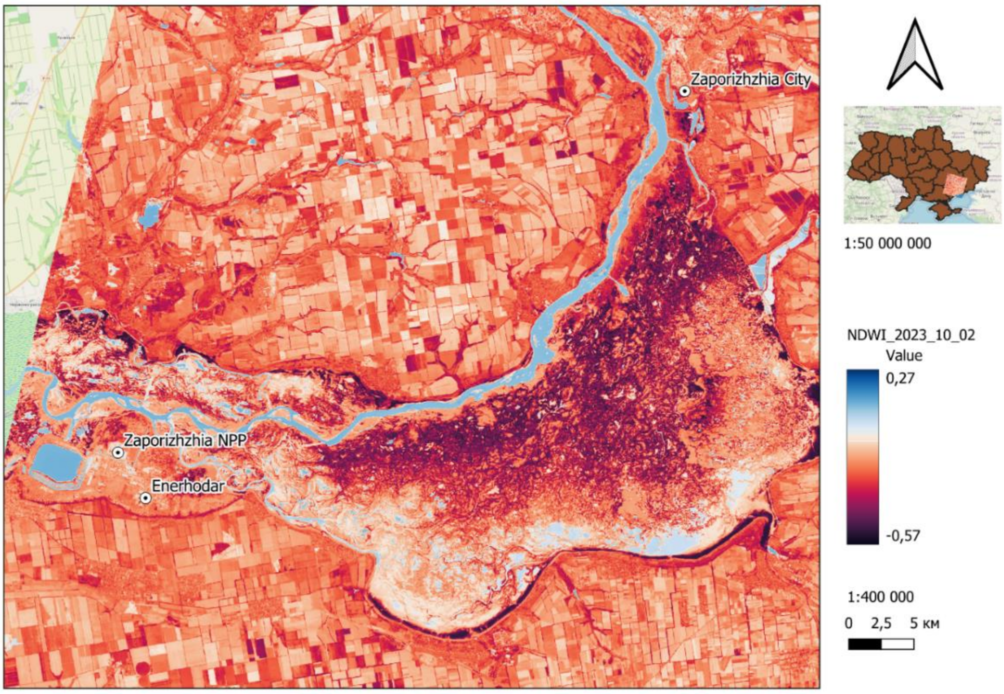


Fig. 1. Visualization of the NDWI index of the Kakhovka reservoir as of 2 October 2023.

The old Dnipro riverbed is observed to remain unchanged in shape, while most shallow water bodies are drying up. The Kakhovka Reservoir has been fragmented into five to eight large reservoirs connected by the old Dnipro riverbed, 15 to 20 medium and large reservoirs that have lost direct connection with the Dnipro, and several hundred medium and small isolated reservoirs. The structural integrity of the dam and the water level in the Zaporizhzhia NPP cooling pond remained unchanged. The Kakhovka Reservoir has been observed to undergo intensive overgrowth of vegetation in the dewatered areas. The environmental safety of the destroyed Kakhovka Reservoir is determined by the location of the Zaporizhzhya NPP industrial site in the zone of its direct hydro-geodynamic impact. This zone is characterized by the presence of quicksand soils, which are sensitive to drainage and hydrogeofiltration compression. The available literature data of design engineering and geological studies indicate that these soils are capable of filtration compaction with deformations of the earth's surface, subsoil, and foundations of structures located on them (reactor compartments, cooling towers, water pipes, and dams). Concerning the cooling pond, and other similar structures, an environmentally, economically, and socially favorable option for restoring the affected region is possible only on a new ideological and technological basis, which excludes the well-known projects of "self-restoration of natural and anthropogenic ecosystems".

Conclusions. The prospects for further research include the continuation of space-based monitoring of ecosystem transformations to predict long-term natural and anthropogenic threats resulting from changes in hydrogeological conditions and to substantiate options for restoring the Kakhovka Reservoir.

1. National Academy of Sciences of Ukraine. (2023). *About the ecological consequences of the destruction of the Kakhovka HPP dam* [PDF]. Retrieved April 10, 2024, from https://files.nas.gov.ua/PublicMessages/Documents/0/2023/09/230915121937204-4928.pdf
2. QGIS Association. (2024). *QGIS geographic information system*. Retrieved April 10, 2024, from http://www.qgis.org
3. U.S. Geological Survey. (n.d.). *Who we are*. Retrieved April 10, 2024, from https://www.usgs.gov/about/about-us/who-we-are

Anton Shykhmat[1], Zenoviy Veres[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, anton.o.shykhmat@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, zenovii.y.veres@lpnu.ua

EVALUATING CLOUD IOT SERVICES: BALANCING TRAFFIC REQUIREMENTS AND COSTS IN AWS AND GCP

Introduction. The Internet of Things (IoT) plays an important role in leading digital transformation in different industries. The seamless exchange of data between IoT devices and cloud platforms is a critical aspect in realizing the full potential of smart systems. This article looks at how businesses can address the balance between the complex traffic needs of IoT systems and the costs involved when using MQTT in HTTP protocols in two of the leading cloud platforms: Amazon Web Services (AWS) and Google Cloud Platform (GCP), and provides a decision tree to select the most optimal protocol and cloud provider.

Materials and methods. Existing studies [1-2] showed that MQTT is 6 times faster, compared to the HTTP protocol, for sending time series data and is more energy-efficient. However, none of the studies take into account the volumes of billed traffic and the costs associated with using the protocols in different usage scenarios. Similarly, these studies do not provide an analysis of the feasibility of using IoT Core solutions from cloud providers such as AWS and GCP [3]. In this research, the billed traffic and total costs were analyzed for the 10 most common data transmission patterns in IoT [4], where different message frequencies were considered, ranging from sending 1Kb messages (minimum billed message size) every minute to every 6 hours, as well as the number of devices that also varied from 10k to 100M.

Cloud providers AWS and GCP have different billing models for IoT Core solutions. On GCP, MQTT connection management (PINGREQ) messages are charged the same way, as data messages, but in AWS such messages are free of charge. On the other hand, AWS charges are based on the total MQTT connection time. The calculations in this research assume that each device connects or reconnects to the MQTT bridge only once per day. For the HTTP bridge, both GCP and AWS charge for every request and response involved in data transmission. The results section presents scenarios that offer the most valuable insights associated with billed traffic and costs.

Results. Scenario 1 - 1Kb message payload, data is transmitted every minute. The data transmission frequency adds a crucial amount of traffic. PINGREQ messages are approximately 5% of all traffic and are relatively small, and their contribution could be ignored for calculations. HTTP bridge uses almost 2x more messages since it is billed for each request and response separately. As a consequence, the HTTP bridge is not applicable for the scenario with high message transmission frequency. HTTP Bridge billed traffic is comparable to MQTT in GCP (for AWS MQTT billed traffic is still twice smaller).

Scenario 5 - 1Kb message payload, data is transmitted every 20 mins. The difference between bridges is caused by the assumption of MQTT connection/reconnection frequency. This assumption adds approximately 0.7% of extra traffic. The main option to decrease the billed traffic cost is to extend the PINGREQ message time from 20 minutes up to the maximum possible value for MQTT according to the specification - 18 hours. It will decrease the PINGREQ traffic 54 times per device. Increasing the PINGREQ time saves almost 49% of billed traffic for the MQTT bridge in GCP and has no changes for AWS. Since each HTTP transmission is billed as two 1Kb messages compared to 1Kb messages for MQTT, the MQTT bridge is a cheaper and preferred approach to transmit the data for scenarios with frequent data transmissions (less than 20 minutes) and stable network connections. The connection traffic shows that MQTT is preferable over the HTTP bridge even if the reconnect needs to happen every 20 minutes – the connection traffic will be equal to the telemetry messages traffic.

Scenario 6 - 1Kb message payload, data is transmitted every 30 mins. Thirty-minute telemetry messages interval is when the GCP MQTT bridge requires 26% more billed traffic than the HTTP ones, but the AWS MQTT bridge requires 48.95% less billed traffic than the HTTP.

Scenario 7 - 1Kb message payload, data is transmitted every hour. MQTT PINGREQ messages add more than 74% of billed traffic to the GCP MQTT bridge. This contribution increases when the message delivery is done less frequently: PINGREQ messages add 83% of billed traffic if a telemetry message is delivered every second hour, and it adds 90% of traffic for scenario 10 (telemetry message is delivered every 6 hours).

Table 1. Billed Traffic (Mb/month) for MQTT and HTTP Bridges for Scenarios 1, 5, 6, 7

	10K devices	100K devices	1M devices	10M devices	100M devices
Scenario 1 AWS MQTT	432.6K	4.326M	43.26M	432.6M	4.326B
Scenario 1 GCP MQTT	453.9K	4.539M	45.39M	453.9M	4.539B
Scenario 1 HTTP	864K	8.64M	86.4M	864M	8.64B
Scenario 5 AWS MQTT	21.9K	219K	2.19M	21.9M	219M
Scenario 5 GCP MQTT	43.5K	435K	4.35M	43.5M	435M
Scenario 5 HTTP	43.2K	432K	4.32M	43.2M	432M
Scenario 6 AWS MQTT	14.7K	147K	1.47M	14.7M	147M
Scenario 6 GCP MQTT	36.3K	363K	3.63M	36.3M	363M
Scenario 6 HTTP	28.8K	288K	2.88M	28.8M	288M
Scenario 7 AWS MQTT	7.5K	75K	750K	7.5M	75M
Scenario 7 GCP MQTT	29.1K	291K	2.91M	29.1M	291M
Scenario 7 HTTP	14.4K	144K	1.44M	14.4M	144M

The billed traffic size is not the only criterion for selecting the solution for connecting IoT devices to the cloud. Expenses are also a crucial point for a business that constraints architecture design. As mentioned earlier, AWS does not charge for PINGREQ messages, but it charges for device connection time. The HTTP bridge could be used for data transmission for 10k up to 1M devices when it happens every 30 minutes or less often. The price comparison shows a massive increase between 1M and 10M devices in GCP. While the device count increased by 10, the GCP MQTT expenses increased 18x for 6 hours of data transmission and almost 20x for 30-minute transmission frequency. The HTTP bridge price is also growing 10x between 1M and 10M devices. For AWS, the situation is different. Price per message decreases when you produce more traffic, and as we can see HTTP protocol is cheaper, even if it produces more billable traffic. Also, we may notice that the AWS MQTT bridge is 3x cheaper than the GCP MQTT bridge, and the AWS HTTP bridge is 6x cheaper than the GCP HTTP bridge. Overall AWS IoT Core is 5 times cheaper than GCP IoT Core for both MQTT and HTTP protocols, and the MQTT bridge in AWS is about 45% cheaper than the HTTP bridge in AWS for frequent data transmission scenarios (<5 mins).

Conclusions. The data transmission depends on the business requirements. The major driver for the proper selection – the telemetry delivery frequency and devices count. As mentioned earlier, GCP IoT Core is more expensive compared to AWS IoT Core for all evaluated scenarios, so it is not recommended for use.

The decision table with recommended options for telemetry delivery is depicted above in Table 2. For almost all cases usage of the AWS IoT Core MQTT bridge is applicable for frequent data delivery.

Table 2. Protocols Selection Decision Tree

Devices count	Messages sent		Data transmission every		Decision
	24 hours per day	8 hours per day	<10mins	>10mins	
<10K	+	-	+	-	AWS MQTT
<10K	+	-	-	+	AWS HTTP
<10K	-	+	+	-	AWS MQTT
<10K	-	+	-	+	AWS HTTP
<100K	+	-	+	-	AWS MQTT
<100K	+	-	-	+	AWS HTTP

<100K	-	+	+	-	AWS MQTT
<100K	-	+	-	+	AWS HTTP
<1M	+	-	+	-	AWS MQTT
<1M	+	-	-	+	AWS HTTP
<1M	-	+	+	-	AWS MQTT
<1M	-	+	-	+	AWS HTTP
<10M	+	-	+	-	AWS MQTT
<10M	+	-	-	+	AWS HTTP
<10M	-	+	+	-	AWS MQTT
<10M	-	+	-	+	AWS HTTP
<100M	+	-	+	-	Custom MQTT broker або CoAP
<100M	+	-	-	+	AWS HTTP
<100M	-	+	+	-	Custom MQTT broker або CoAP
<100M	-	+	-	+	AWS HTTP
>100M	+	-	+	-	Custom MQTT broker або CoAP
>100M	+	-	-	+	AWS HTTP
>100M	-	+	+	-	Custom MQTT broker або CoAP
>100M	-	+	-	+	AWS HTTP

When device count increases (>10M) and data is delivered at a high frequency (approx. every 1 minute) it is recommended to use either a standalone MQTT broker or explore other TCP-based protocols, e.g., CoAP [5]. For non-frequent data transmissions, HTTP-bridge could be a solution for up to 100M devices. Considering the IoT Use Case Adoption Report 2021 [6], MQTT is a good choice for remote asset monitoring, vehicle fleet management, location tracking, and track & trace use cases. HTTP is a good choice for IoT-based process automation and predictive maintenance use cases.

1. Atmoko, R. A. A., Riantini, R., & Hasin, M. K. H. (2017). IoT real-time data acquisition using MQTT protocol. *Journal of Physics: Conference Series, 853,* 1-6. https://doi.org/10.1017/9781108913560
2. Wukkadada, B., Wankhede, K., Nambiar, R., & Nair, A. (2018). Comparison with HTTP and MQTT in Internet of Things (IoT). In *2018 International Conference on Inventive Research in Computing Applications (ICIRCA)* (pp. 249-253). IEEE. https://doi.org/10.1109/ICIRCA.2018.8597401
3. Pierleoni, P., Concetti, R., Belli, A., & Palma, L. (2019). Amazon, Google, and Microsoft solutions for IoT: Architectures and a performance comparison. *IEEE Access.* https://doi.org/10.1109/ACCESS.2019.2961511
4. Amadeo, M., Campolo, C., Iera, A., & Molinaro, A. (2015). Information centric networking in IoT scenarios: The case of a smart home. In *2015 IEEE International Conference on Communications (ICC)* (pp. 648-653). IEEE. https://doi.org/10.1109/ICC.2015.7248395
5. Alhaj, A. (2018). Constraint application protocol (CoAP) for the IoT. Frankf. Univ. Appl. Sci. https://doi.org/10.13140/RG.2.2.33265.17766
6. IoT Analytics. (2021). *IoT use case adoption report 2021*. Retrieved from https://iot-analytics.com/product/iot-use-case-adoption-report-2021-2/

Victor Sineglazov[1], Kirill Riazanovskiy[2]
[1]National Aviation University, Kyiv, Ukraine, svm@nau.edu.ua
[2]National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine, k.riazanovskyi@kpi.ua

SEMI-SUPERVISED APPROACH FOR MRI BRAIN TUMOR SEGMENTATION USING 4D ATLAS PRIORS AND GAUSSIAN MODELS

Introduction. According to recent studies [1-2], brain cancer remains a significant global health concern, accounting for 1.9% of all cancer cases and 2.5% of cancer-related deaths worldwide. In 2019, 347,992 new cases were reported, with higher incidence rates in males (54%) compared to females (46%). Glioblastoma, the most prevalent malignant brain tumor, comprises approximately 49% of all malignant cases. Despite therapeutic advancements, the prognosis for glioblastoma remains poor, with a five-year relative survival rate increasing only slightly from 4% in the mid-1970s to 7% in recent years [2].

Magnetic resonance imaging (MRI) plays a crucial role in diagnosis, but segmenting MRI images to identify tumors is complex due to the unique characteristics of brain tumors. The primary task of MRI brain image processing is the segmentation task, which is efficiently solved by deep neural networks. The main problem in training such a network, whether it is a convolutional neural network or a vision transformer, is the availability of a sufficiently labeled training sample. Unfortunately, in real-world situations, the samples are limited and have a small number of labeled scans since high-quality labeling requires availability and a huge amount of time from a qualified medical radiologist, which is not always possible.

To address the problem of insufficient sample and limited resources for labeling and training the model, there are different approaches, the most popular ones are:

- transfer learning – transferring knowledge from a more general known dataset to a specific limited dataset,
- active learning – methods for identifying the most relevant data for manual partitioning by a specialist,
- semi-supervised learning – model training using unlabeled data, among others.

Each of the approaches deserves individual attention, but this paper proposes the use and improvement of semi-supervised learning as one of the most promising approaches using unlabeled sampling.

Problem statement. The work is focused on developing a semi-supervised learning method for brain image segmentation using 4D atlas priors, allowing efficient use of unlabeled data to improve segmentation accuracy. The objective is to create a semi-supervised learning method that utilizes 4D atlas priors for brain tumor segmentation on MRI images, enhancing model accuracy and generalization with a limited amount of labeled data. For the labeled sample and the unlabeled sample, create a classifier that correctly predicts the binary mask of the new scan tensor utilizing both samples.

Materials and methods. The method is based on constructing a probabilistic 4D atlas that includes the coordinates and voxel intensities of labeled segments and generalizing this atlas using Gaussian Mixture Models (GMM). The atlas is utilized in three forms: as a loss function, for generating pseudomasks, and for pseudomask validation. The performance of the proposed method was tested on real MRI data with axial brain tumor images.

A private MRI dataset of brain tumors was used for experiments and validation. The T1 modality and axial view were employed. The dataset comprised 34 patients and 1144 images. Labeling was performed manually by medical professionals with over 10 years of experience.

To account for anatomical structures and the natural location of anomalies, as well as their color, we propose creating 4D atlas priors based on labeled data segments. The first three dimensions of the atlas are the coordinates of the anomaly location in a 3D scan. The fourth dimension is the observed color or contrast of anomaly pixels.

To generalize the atlas and remove input data limitations, we propose modeling the pixel distributions of the given atlas using GMM. Modeling through GMM helps the models better generalize to new data without being restricted to the existing tumor atlas, as GMM assigns the probability of tumor presence to a wider range of pixels.

We propose using the negative log-likelihood (NLL) of the data under the GMM model as a loss function based on the 4D GMM atlas.

Three training options based on the proposed 4D atlas are suggested:

A new loss function that maximizes the likelihood of the location and voxel intensities of unlabeled image pseudomasks and atlas values in the region of the resulting pseudomask.

A loss function based on location likelihood only, but using MSE for the atlas intensities.

Pseudomask validation for atlas-based sample expansion.

Results. The results demonstrated (Table 1) that the proposed semi-supervised learning methods based on the atlas achieved IoU nearly similar to training on the full dataset. The best-performing method was the 3D GMM loss + voxel intensity loss, achieving an accuracy of 0.9 with only 30% of the data used.

Table 1. Resulting IoU accuracy of the proposed approach on the test sample

Method/Percentage of data used	5%	10%	30%	50%
4D GMM atlas loss	0.66	0.76	0.88	0.9
3D GMM loss + voxel intensity loss	**0.68**	**0.79**	**0.9**	**0.92**
Pseudomasks validation	0.62	0.75	0.87	0.89
Lee et al. [3]	0.55	0.68	0.8	0.82
Clough et al. [4]	0.54	0.66	0.79	0.81

Conclusion. In this paper, we introduced an advanced semi-supervised learning framework for brain tumor segmentation that leverages 4D atlas priors to utilize both labeled and unlabeled data effectively. Our approach constructs a probabilistic 4D atlas based on labeled segments, generalizes this atlas using GMM, and incorporates additional dimensions to account for contrast variations within the anomalies.

The proposed method addresses the common limitations found in existing research, which often focus on either shape priors, atlas priors, or semi-supervised learning in isolation. By integrating the generalized localization of anomalies, their contrast, and precise boundary delineation into a single framework, we achieve a more comprehensive and robust segmentation solution.

1. Ilic, I., & Ilic, M. (2023). International patterns and trends in the brain cancer incidence and mortality: An observational study based on the global burden of disease. *Heliyon*, *9*(7), e18222. https://doi.org/10.1016/j.heliyon.2023.e18222
2. Miller, K. D., Ostrom, Q. T., Kruchko, C., Patil, N., Tihan, T., Cioffi, G., Fuchs, H. E., Waite, K. A., Jemal, A., Siegel, R. L., & Barnholtz-Sloan, J. S. (2021). Brain and other central nervous system tumor statistics, 2021. *CA: A Cancer Journal for Clinicians*, *71*(5), 381-406. https://doi.org/10.3322/caac.21693
3. Lee, M. C. H., Petersen, K., Pawlowski, N., Glocker, B., & Schaap, M. (2019). TeTrIS: Template Transformer Networks for Image Segmentation With Shape Priors. *IEEE Transactions on Medical Imaging*, *38*(11), 2596–2606. https://doi.org/10.1109/TMI.2019.2905990
4. Clough, J. R., Byrne, N., Oksuz, I., Zimmer, V. A., Schnabel, J. A., & King, A. P. (2019). A topological loss function for deep-learning-based image segmentation using persistent homology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *44*, 8766–8778.

Victor Sineglazov[1,2], Alexander Ruchkin[3]
[1]National Aviation University, Kyiv, Ukraine, svm@nau.edu.ua
[2]Institute of Cybernetics of the National Academy of Sciences of Ukraine, Kyiv, Ukraine, svm@nau.edu.ua
[3]National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine, alex3005r@gmail.com

MODIFICATED GRAPH-BASED SEMI-SUPERVISED LEARNING METHOD BASED ON THE POISSON'S EQUATION

Introduction. Modern semi-supervised learning (SSL) techniques are increasingly utilized to address data classification challenges, especially when faced with a vast quantity of unlabeled data and a limited subset of labeled data. Graph-Based Semi-Supervised Learning (GSSL) distinguishes itself among SSL approaches with its notable advantages, such as enhanced performance when dealing with minimal amounts of unlabeled data and its adeptness at capturing intricate relationships. This makes it particularly suited for challenging datasets. The application of graph-based SSL extends across various domains of data analysis, including but not limited to, healthcare, finance, meteorology, and archaeology, showcasing its versatility and effectiveness. In the GSSL algorithm, label computation is exclusively performed on the graph's vertices, rather than across all vertices. This approach significantly reduces the algorithm's operational complexity, enhancing its efficiency. Currently, semi-supervised learning based on graphs underlies the construction of various Hybrid Neural Networks that are used in medical diagnostics [1-3].

The foundational technique for label propagation is the "Label Propagation (Laplace Label Propagation)" method, introduced by [4]. This method enables the diffusion of known labels throughout the graph, allowing for the estimation of labels for unobserved nodes based on their closeness and connections to labeled ones with the help of Laplace equation. This strategy not only leverages the limited available labeled data but also efficiently harnesses the vast amounts of unlabeled data. The advancement of this field is seen as highly promising and impactful for addressing numerous practical challenges.

A contemporary and more advanced approach to semi-supervised learning is the " Poisson Learning (Poisson Label Propagation)", first proposed [7], with subsequent mentions in works by [8]. From the perspective of SSL, Poisson learning bears a resemblance to Laplace learning; however, it utilizes Poisson's equation for propagating labels. In this paper, we will focus on implementing modified Poisson training to explore its potential and efficacy within the realm of GSSL.

Materials and methods. Currently, there are many different approaches for implementing semi-learning with Laplacian, which can be viewed in the reviews [4-7, 10-12]. Graph-Based Semi-Supervised Learning methods are often used to solve classification problems GSSL faces several key challenges at different stages. Here are some of the main problems and methods used to solve them in GSSL.

Label propagation stage. The main purpose of GSSL is to propagate label values under poorly labeled sample conditions (Low Labeled Dataset) [4-6]. However, the information from which labels derive their value may be incorrect, which in turn leads to wrong label values, which worsens classification results. To solve this problem, various algorithms have been developed, such as label diffusion [8]. These algorithms use the structure of the graph to form labels while taking into account the overlap between data points.

Stage of processing a limited amount of labeled data: GSSL aims to use both labeled and unlabeled data. However, when labeled data is insufficient, methods such as self-learning, collaborative learning, and active learning can be used [7-8]. These methods iteratively select informative samples for labeling or use multiple presented data to enhance the learning process.

Optimization. Often, an optimization problem can be represented as a target function to be optimized and a certain set of initial data and restrictions on the solution. The problem of optimizing the label distribution process is one of the important problems of semi-learning [11]. Usually, it is enough to use the gradient method, but in the case of large data, it is better to use faster and more stable methods for finding solutions, for example, the ADAM method.

Let's note some of the advantages of Poisson Label Propagation over the classical Laplace Label Propagation. First of all, it means higher speed and shorter execution time due to the use of precise differential methods rather than stochastic methods (Random Walk, Diffuse, Broun Move). Secondly, it is overcoming degeneracy at zero – i.e., when the number of specified labeled vertices is very small with respect to all vertices – or the ratio of labeled vertices to all vertices is zero. Third, it is overcoming the problem of forgetting a large amount of data. That is, in Laplace learning, with a small number of initial labeled vertices and a large number of unlabeled vertices, it is possible to relabel their initial values.

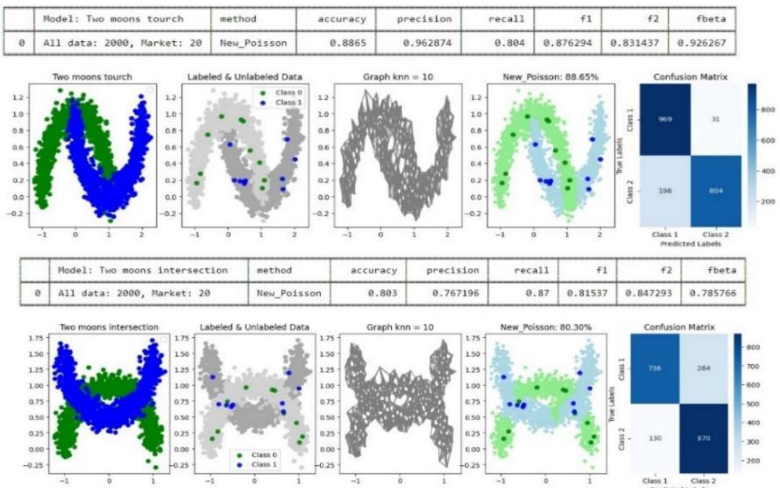
Modified Poisson Label Propagation. Let be given an undirected weighted graph *G* = *G*(*X*, *V*, *W*) with n vertices. $X = \{x_1, x_2, \dots, x_n\}$ is the set of vertices of the graph, $V = \{v_1, v_2, \dots, v_n\}$ is the set of edges, $W = (w_{ij})_{i,j=1}^{n}$, – the matrix of weights of the graph *G*. Let us put w_{ij} = 1 if the vertices x_i, x_j are similar, and w_{ij} = 0, if the vertices x_i, x_j are different. The degree of the node is determined by the formula $d_i = \sum_{j=1}^{n} w_{ij}$. Proposed modified Poisson equation with initial conditions $u_i = y_i$, $1 \le i \le m$:

$$L(u_i) = \sum_{j=1}^{m} (y_j - \underline{y})\delta_{ij} + \alpha_1 \|u\|, \qquad m+1 \le i \le n,$$

where *L* is the non-normalized Laplace operator, x_i is the vertices of an undirected weighted *G* graph, $y_i = y(x_i)$ is the initial marks of the vertices of the graph, $u_i = u(x_i)$ is the function of the vertex labels of the graph, *n* is the total number of vertices of the graph, *m* – labeled. $\delta_{ij} - Kronecker's\ symbol.\ \underline{y} = \frac{1}{m}\sum_{j=1}^{m} y_j.\ \sum_{j=1}^{n} d_i u(x_i) = 0.$

The problem of algorithm convergence rate for big data remains. There are many optimization algorithms to improve the solution convergence problem. One of the best optimization methods is the ADAM method, which we will apply next [11]. In this paper, we will consider the method of semi-supervised learning using the modified Poisson equation and additional regularization for more effective application on uncomfortable samples with the ADAM method.

Results. Examples of synthetic data. There is the task classification of two moons without intersection and with intersection data. Here are the results of calculating the following metrics for the proposed method: accuracy, precision, recall, f1, f2, fbeta, which are summarized in the table. We consider two cases of two months (with and without class overlap) with a total of 2000 data points and labeled 20 by 10 in each class. Figure 1 shows the given classes, initial labels, graph construction using the KNN-10 method, classification results, and the confusion matrix.



	Model: Two moons tourch	method	accuracy	precision	recall	f1	f2	fbeta
0	All data: 2000, Market: 20	New_Poisson	0.8865	0.962874	0.804	0.876294	0.831437	0.926267

	Model: Two moons intersection	method	accuracy	precision	recall	f1	f2	fbeta
0	All data: 2000, Market: 20	New_Poisson	0.803	0.767196	0.87	0.81537	0.847293	0.785766

Fig. 1. Results of calculating metrics with and without class intersection

Conclusion. This paper explores the training of neural networks through a novel approach based on the Poisson equation, incorporating L2 regularization. We introduce a new generalized formula for Poisson learning. This approach leverages the relatively recent machine learning paradigm of Poisson learning, achieving superior outcomes compared to the traditional Laplace teaching method. The proposed approach allows the accuracy of solving medical diagnostics tasks to be increased by using a small amount of labeled data while using a large amount of unlabeled data. The calculated accuracy value (>80%) in the most difficult case of class intersection makes this method useful for use in practical problems of medical diagnostics.

1. Sineglazov, V., Riazanovskiy, K., Klanovets, A., Chumachenko, E., & Linnik, N. (2022). Intelligent tuberculosis activity assessment system based on an ensemble of neural networks. *Computers in Biology and Medicine*, 147, Article 105800. https://doi.org/10.1016/j.compbiomed.2022.105800
2. Sineglazov, V., & Kot, A. (2021). Design of hybrid neural networks of the ensemble structure. *Eastern-European Journal of Enterprise Technologies*, 1(4), 31–45. https://doi.org/10.15587/1729-4061.2021.225301
3. Zgurovsky, M., Sineglazov, V., & Chumachenko, E. (2021 Formation of hybrid artificial neural networks topologies. In *Artificial Intelligence Systems Based on Hybrid Neural Networks* (Vol. 904, pp. 1–11). Springer, Cham. https://doi.org/10.1007/978-3-030-48453-8_3
4. Zhu, X. (2005). Semi-supervised learning literature survey (TR 1530). University of Wisconsin-Madison. https://minds.wisconsin.edu/handle/1793/60444
5. Van Engelen, J. E., & Hoos, H. H. (2020). A survey on semi-supervised learning. *Machine Learning*, 109, 373–440. https://doi.org/10.1007/s10994-019-05855-6
6. Song, Y., Zhang, J., & Zhang, C. (2022). A survey of large-scale graph-based semi-supervised classification algorithms. *International Journal of Cognitive Computing in Engineering*, 3, 188–198. https://doi.org/10.1016/j.ijcce.2022.10.002
7. Song, Z., Yang, X., Xu, Z., & King, I. (2022). Graph-based semi-supervised learning: A comprehensive review. *IEEE Transactions on Neural Networks and Learning Systems*. https://doi.org/10.1109/TNNLS.2022.3155478
8. Yang, X., Song, Z., King, I., & Xu, Z. (2022). A survey on deep semi-supervised learning. *IEEE Transactions on Knowledge and Data Engineering*. https://doi.org/10.1109/TKDE.2022.3220219
9. Calder, J., Cook, A. B., Thorpe, M., & Slepcev, D. (2020). Poisson learning: Graph based semi-supervised learning at very low label rates. In *Proceedings of the 37th International Conference on Machine Learning* (Vol. 119, pp. 1306–1316). PMLR. https://proceedings.mlr.press/v119/calder20a.html
10. Aromal, M. A., & Rasool, A. (2021). Semi-supervised learning using graph data structure: A review. In *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)* (pp. 894–899).
11. Llugsi, R., Yacoubi, S. E., Fontaine, A., & Lupera, P. (2021). Comparison between Adam, AdaMax, and AdamW optimizers to implement a weather forecast based on neural networks for the Andean city of Quito. In *2021 IEEE Fifth Ecuador Technical Chapters Meeting (ETCM)* (pp. 1–6).
12. Belkin, M., Niyogi, P., & Sindhwani, V. (2006). Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7, 2399–2434. http://jmlr.org/papers/v7/belkin06a.html

Viktor Syneglazov[1], Kyrylo Lesohorskyi[2], Olena Chumachenko[3]
[1]*National Aviation University, Kyiv, Ukraine, svm@nau.edu.ua*
[2]*Kyiv Polytechnic Institute, Kyiv, Ukraine, lesogor.kirill@gmail.com*
[3]*Kyiv Polytechnic Institute, Kyiv, Ukraine, eliranvik@gmail.com*

SEMI-SUPERVISED LABEL PROPAGATION BASED ON SMOOTHNESS AUGMENTED CONSENSUS

Introduction. Semi-supervised learning (SSL) techniques combine the advantages of supervised and unsupervised learning, allowing the use of unlabeled data to improve model performance. One such method is an algorithm called "label propagation" (Label Propagation) [1], which uses a graph data structure to propagate labels from labeled to unlabeled points.

The label propagation algorithm is built on the idea of creating a graph, where nodes represent data points and edges represent similarities between them. Labels are propagated through this graph, using similarity information to classify unlabeled points. A key parameter in this method is σ, which controls the weight of edges in the graph and affects how far labels can extend. The label propagation algorithm is made possible by applying smoothness and manifold assumptions to the data.

One of the key assumptions of semi-supervised learning is the smoothness assumption. It states that if two points, x_1 and x_2 are close in the input space, their labels y_1 and y_2 should be the same. As such, the effectiveness of algorithms based on the smoothness assumption is heavily impacted by the distance metric [2]. However, in practice selecting the distance metric is an extremely challenging task, especially for higher dimensionality datasets.

This work considers a hybrid approach to inductive semi-supervised learning, which combines the label propagation algorithm (Label Propagation) and the support vector method (SVM) [3] to create an implicit hybrid distance metric based on consensus.

Problem Statement. In this work, we consider a classical semi-supervised learning setting in a multi-class classification scenario. In this setting, dataset D is split into labeled samples $L = \{(x_1, y_1), (x_2, y_2), \dots (x_n, y_n)\}, x \in X, y \in Y$, where X is an input space, Y is a label space, n is the number of labeled samples, and unlabeled samples $U = \{x_{n+1}, x_{n+2}, \dots, x_{n+m}\}$, where m is the number of unlabeled samples. Given a classifier function $f_S: R^D \to \{1, \dots C\}$ trained using a known supervised algorithm, where C is the number of classes, the goal is to build a semi-supervised classifier function $f_{SSL}: R^D \to \{1, \dots C\}$, which outperforms supervised classifier.

Depending on the problem, different desirable metrics can be chosen. In this work, we use an F-score as it captures both accuracy and specificity of the classifier:

$$F_C = 2 * \frac{precision \cdot recall}{precision + recall} = \frac{TP_C}{TP_C + \frac{1}{2}(FP_C + FN_C)}, \quad (1)$$

where *TP* is a true positive count for class *c*, *FP* is a false positive count for class *c*, *FN* – is a false negative count for class *c*.

Semi-supervised learning itself can be considered an optimization problem of objective (loss) function:

$$L(f_{SSL}) = L_L(f(X_L), Y_L) + \alpha L_U(f(X_U)), \quad (2)$$

where L_L is the labeled loss component that utilizes both input space and label space, L_U is the unlabeled loss component that utilizes only input space, and α is the regularization hyperparameter that controls the contribution of a semi-supervised component to overall loss.

Method. The proposed method is an augmentation of the existing graph-based SSL method [4] with a smoothness assumption. The key disadvantage of the existing method is that it relies on the hyperparameter σ to control the distance scaling, or how quickly the weight of each graph edge decreases. It is assumed that data density is uniform, however, in real-world scenarios data density can differ from class to class and within the input space itself. As such, selecting σ correctly is extremely important, however, it is a tedious process that needs to be repeated for each dataset. In this work we aim to improve the algorithm by making σ a local parameter that is computed automatically, improving the adaptability and versatility of the algorithm. To achieve this, each vertex in the graph is assigned a local parameter σ_l based on its neighborhood to account for local density fluctuations.

The first step of the algorithm involves building a fully connected graph $G = (V, E)$, where V is the set of vertices formed from both labeled and unlabeled dataset $D = L \cup D$ with each vertice representing a point in the input space $x_i \in X$, E is the set of edges, each edge represents similarity (or distance) between two points in the input space.

To build the graph, all of the input samples are combined to form a set of vertices. Then for each vertice x_i, a locality density measure σ_l^i is calculated using a k-nearest neighbors method:

$$\sigma_l^i = \frac{1}{k}\sum_{j\in NN(x_i)} |x_i - x_j|, \quad (3)$$

where $NN(x_i)$ are nearest neighbors based on the distance metric $d = |x_i - x_j|$ is the distance metric used to measure similarity, in this work, we use Euclidean distance, which can be changed depending on the dataset, k is the hyperparameter of how many neighbors are used to calculate the locality. After the locality density measure is calculated for all vertices, the next step involves calculating an edge weight between all of the pairs of vertices as:

$$w_{ij} = e^{-\frac{|x_i - x_j|^2}{2\sigma_l^i\sigma_l^j}}, \quad (4)$$

where $|x_i - x_j|$ is the distance metric used to measure similarity. After this step, the matrix $W^{m+n\times m+n}$ is formed, finalizing the graph construction process.

The second step of the algorithm involves converting the graph into a more computationally-friendly format. This is done by translating the graph G into transition probability matrix T as:

$$T_{ij} = \frac{w_{ij}}{\sum_{k=1}^{m+n} w_{ik}}, \quad (5)$$

where T_{ij} represents the probability of translation between the vertices x_i and x_j.

After this, the preparation is complete and the training loop begins. First, an output matrix $Y^{(n+m)\times C}$ is formed, with each row Y_i representing the label probability distribution for the sample x_i. Initial values for this matrix are set either randomly or by a uniform distribution.

After the matrix is created, it is properly initialized. For each known label, a one-hot encoded vector is set into corresponding rows, which represents high confidence in the selected samples.

After the matrix is created and initialized, an iterative training loop begins [5]. On each iteration of the loop the following sub-steps are executed:

1. supervised training is performed using labeled sample L;
2. trained classification function is then used to create the updated prediction matrix Y';
3. labels are propagated based on the transition probability matrix as $Y \leftarrow T \cdot Y$, ground truth labels are skipped during the process;
4. weights are re-normalized as $Y_{ij} \leftarrow \frac{Y_{ij}}{\sum_{k=1}^{C} Y_{ik}}$;
5. high-confidence samples H (samples where the confidence of one of the classes is above the threshold p) are promoted to labeled dataset L to be used during the training process, or $L = L \cup H, H \in U \mid p \leq Y_H$.

The process is repeated until the process converges. After the process concludes, the residual labels are assigned as $\hat{y}_i = arg\ arg\ Y_{ij}$ and the final training round is performed.

Results. The proposed algorithm was tested on a synthetic two-moon dataset, with a slight augmentation – instead of two moon crests we use four moon crests arranged in the same pattern to introduce extra classes. Each crest has 100 samples, but also a slightly different density parameter to ensure non-equal distribution and density differentiation. The dataset is then converted into a semi-supervised dataset by removing labels from some of the input samples in an 80-20 ratio. The algorithm was tested with a support vector machine and random forest classifiers. The test results are compared against the baseline algorithm with different sigma settings. The results are presented in Table 1.

Table 1. Results of the Proposed Algorithm on the Synthetic Two-Moons Dataset

Classifier	**Sigma 0.1**	**Sigma 0.2**	**Sigma 0.5**	**Adaptive Graph**
SVM	0.9498	0.9350	0.8571	**0.9503**
RandomForestClassifier	0.9167	0.9292	0.9214	**0.9320**

It is also worth noting how changing some of the parameters will affect the efficiency of the algorithm. Two parameters that affect performance the most are distance metric d and neighbor count k. In this work, we use $k = 5$ and Euclidean distance d. In our case, a dataset is a collection of two-dimensional pairs of coordinates, which favors Euclidean distance greatly. As the count of dimensions increases, this leads to the "curse of dimensionality" and a rapid drop in the method's efficiency. A number of neighbors k controls how many closest points are considered to determine the true class of the label. k should be treated as a hyperparameter and can be tuned during the training. Setting k too small leads to suboptimal performance in high-density regions with class overlap. If k is too big, this leads to suboptimal label propagation as the locality gets diluted and confidence in each sample drops.

Conclusion. This work presents an augmentation of the graph algorithm with smoothness-based locality consensus, which improves the performance of the underlying training algorithm. The algorithm requires a few hyperparameters that are easy to tune and can adapt to variations in the input space density.

Future work focuses on the expansion of the distance metrics that can be used and tuning them depending on the specificity of the underlying training dataset, especially in challenging scenarios with high-dimensional data, like images. The Euclidean distance metric works well on the synthetic dataset with a small dimensionality, but performance drops as data diversifies and dimensionality increases. Alternative distance metrics, such as other lp norms, cosine similarity, and autoencoder-based distance calculations should be considered.

1. Iscen, A., Tolias, G., Avrithis, Y., & Chum, O. (2019). Label propagation for deep semi-supervised learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 5070-5079).
2. Li, Z., Kang, Y., Lv, W., Zheng, W. X., & Wang, X. M. (2020). Interpretable semisupervised classification method under multiple smoothness assumptions with application to lithology identification. *IEEE Geoscience and Remote Sensing Letters*, *18*(3), 386-390.
3. Govada, A., Joshi, P., Mittal, S., & Sahay, S. K. (2015). Hybrid approach for inductive semi supervised learning using label propagation and support vector machine. In *Machine Learning and Data Mining in Pattern Recognition: 11th International Conference, MLDM 2015, Hamburg, Germany, July 20-21, 2015, Proceedings 11* (pp. 199-213). Springer International Publishing.
4. Blum, A., & Chawla, S. (2001). Learning from Labeled and Unlabeled Data using Graph Mincuts. *International Conference on Machine Learning*, 19–26.
5. Zgurovsky, M. Z., Sineglazov, V. M., & Olena, I. C. (2020). *Artificial Intelligence Systems Based on Hybrid Neural Networks* (Vol. 390). Springer.

Olesia Tehlivets[1], Olha Narushynska[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, olesia.tehlivets.knm.2020@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, narushynska@gmail.com

MODEL AND ANALYTICAL TOOLS FOR INVESTIGATING TEENAGERS' SLEEP DISORDERS USING ACCELEROMETER DATA

Introduction. Sleep disorders in adolescents are increasingly recognized for their negative impacts on health, education, and psychological well-being. Traditional monitoring methods are often intrusive and impractical for long-term use, highlighting the need for innovative approaches. This research proposes the use of wrist-worn accelerometers to monitor sleep patterns in a less intrusive manner, applying machine learning techniques to process and analyze the data effectively.

The main goal of the project is to design and develop a system for analyzing sleep periods based on accelerometer data for monitoring children's/adolescents' sleep. Based on the dataset provided by CMI, it is necessary to determine when a child falls asleep and when he or she wakes up.

The object of research is developing a system for detecting the sleep periods of adolescents, which are determined by analyzing data from an accelerometer worn on the wrist. This data explores human movements and activity during sleep and wakefulness.

The subject area of this project focuses on the automatic detection of sleep and wakefulness periods in adolescents using accelerometer data. This task is significant for several reasons, primarily related to health monitoring and preventing sleep-related issues. Accurately identifying sleep patterns can aid in diagnosing sleep disorders, improving overall health, and optimizing daily performance.

Materials and methods. The study utilized accelerometer data from wrist-worn devices, employing machine learning algorithms, including Long Short-Term Memory (LSTM) networks, to analyze sleep patterns. Data preprocessing involved normalization and feature extraction, crucial for the subsequent model training and evaluation phases.

The developed system offers a scalable and non-intrusive method for monitoring sleep patterns, providing a tool for healthcare providers to diagnose and treat sleep disorders in adolescents effectively. Its application could lead to improved health outcomes and better understanding of adolescent sleep behavior.

The methodology involves several key stages:

- Data Collection: Collecting accelerometer data from wearable devices.
- Data Preprocessing: Cleaning and preparing the data for analysis, including feature extraction.
- Model Implementation: Employing machine learning algorithms to develop models capable of predicting sleep and wake periods.
- Evaluation: Assessing the models using various performance metrics.
- System Development: Integrating all components into a comprehensive software system.

The dataset for study comes from a competition hosted by the Child Mind Institute and Kaggle. It includes accelerometer readings and sleep diary entries from 277 teenagers aged 11-14 with various sleep disorders. The schema of the data is presented on Figure 1. This dataset was crucial for training and evaluating the model.

Fig. 1. Data structure

Multiple machine learning (ML) models, including Logistic Regression, Decision Tree, Random Forest, Gradient Boosting, and Long Short-Term Memory (LSTM) neural networks, were developed and evaluated (Fig. 2).

Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	0.915854	0.950593	0.920590	0.935351
Decision Tree	0.884079	0.887181	0.944839	0.915102
Random Forest	0.947054	0.969048	0.950280	0.959572
Gradient Boosting	0.916731	0.925219	0.950926	0.937897

Fig. 2. ML models metric

Among these, the LSTM model demonstrated the highest performance in detecting sleep and wakefulness periods. It achieved a high accuracy and a superior F1-score, indicating its effectiveness in handling time-series data from accelerometers.

```
        Classification report for LSTM

              precision    recall  f1-score   support

         0.0       0.90      0.96      0.93    700188
         1.0       0.98      0.94      0.96   1366632

    accuracy                           0.95   2066820
   macro avg       0.94      0.95      0.94   2066820
weighted avg       0.95      0.95      0.95   2066820
```

Fig. 3. LSTM metrics

The models were evaluated using various metrics such as accuracy, precision, recall, F1-score, and Area Under the Precision-Recall Curve (AUC PR). The LSTM model, in particular, excelled with an AUC PR of 0.96, reflecting its robust ability to differentiate between sleep and wake states (Fig. 3).

The overall structure of the software project. The overall structure of the software project for identifying sleep periods in adolescents using accelerometer data is designed to be modular and efficient. The system is divided into several key components, each responsible for specific tasks.

First, the User Authorization Module handles the registration and authentication of users, ensuring secure access to the system. Once authorized, users can upload their accelerometer data.

Next, the Data Preprocessing Module takes the raw data and prepares it for analysis. This involves cleaning the data to remove any errors, transforming it into a usable format, and extracting essential features that will help to predict sleep and wake periods.

The Classification Module then uses this preprocessed data to predict whether the individual is asleep or awake. This is done using various machine learning models, including traditional models like Random Forest and more advanced ones like LSTM (Long Short-Term Memory) neural networks.

After the initial predictions, the Post-processing Module refines these results to ensure they meet specific criteria for valid sleep periods. This helps to smooth out any short interruptions in sleep that might otherwise be misclassified.

All data, including the original data, processed data, and predictions, are managed by the Database Management Module. This module uses a database to store and organize the information, making it easy to retrieve and analyze later.

The User Interface Module provides a web application where users can interact with the system. This interface allows them to upload data, view results, and access reports.

Finally, the Reporting and Visualization Module generates reports and visualizations based on the analyzed data. These tools help users to quickly understand the results and gain insights into their sleep patterns.

Overall, the project is designed to be flexible and scalable. New features or models can be added without disrupting the existing system. The system is built to handle increasing data efficiently, ensuring it can grow and adapt to future needs. This structured approach ensures that the project meets its goals and provides a reliable tool for monitoring and analyzing sleep.

Conclusions. The research successfully developed a reliable system for detecting sleep and wake periods in adolescents using accelerometer data, marking a significant advance in the field of sleep monitoring. Future work will focus on refining the models and expanding their applicability to other demographic groups.

In addition, the use of sleep period filtering significantly improved the classification results. The short periods of activity that used to break up sleep were smoothed out, resulting in more coherent and stable sleep periods. This improved the overall accuracy and adequacy of the classification results by reducing the impact of short-term interruptions.

Thus, the developed models and software have great potential for further research in the field of sleep and practical application in clinical practice. They contribute to a deeper understanding of sleep patterns and create the prerequisites for developing new effective adolescent insomnia treatments.

1. Chase, J. D., Busa, M. A., Staudenmayer, J. W., & Sirard, J. R. (2022). Sleep measurement using wrist-worn accelerometer data compared with polysomnography. *Sensors*, *22*(13), 5041. https://doi.org/10.3390/s22135041
2. Byun, J.-H., Kim, K. T., Moon, H.-j., Motamedi, G. K., & Cho, Y. W. (2019). The first night effect during polysomnography, and patients' estimates of sleep quality. *Psychiatry Research*, *274*, 27–29. https://doi.org/10.1016/j.psychres.2019.02.011
3. Bhaskar, S., Hemavathy, D., & Prasad, S. (2016). Prevalence of chronic insomnia in adult patients and its correlation with medical comorbidities. *Journal of Family Medicine and Primary Care*, *5*(4), 780–784. https://doi.org/10.4103/2249-4863.201153/

Vasyl Teslyuk[1], Stanislav Ivasiv[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, vasyl.m.teslyuk@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, stanislav.s.ivasiv@lpnu.ua

INCREASING MACHINE LEARNING MODEL EFFICIENCY FOR CLASSIFYING CLOTHING PATTERN ELEMENTS

Introduction. Pattern recognition on clothing items is a relevant and important aspect for many fields, including fashion, manufacturing, design, and technology.

The main task of the research is the process of classifying the patterns of clothing elements, which depends on the selected architecture of the neural network, its hyperparameters, and the applied approaches to learning the neural network. The purpose of the work is to research and implement models and tools for increasing the efficiency of the classification of patterns of clothing elements using machine learning.

Research materials and methods. The research used the object-oriented approach, the method of machine learning – learning with the teacher, and data structures. When applying artificial neural networks, the principles of convolutional neural networks, current architectures of convolutional neural networks, and separate sections of math are used.

Analysis of recent research and publications. The work [1] describes methods of background removal from images, outlining their functionality and areas of application. The work [2] outlines methods for locating and classifying clothing in images, but the Clarifai platform lacks access to model architecture and the ability to modify or train models on custom data. The work [3], which describes the problems and solutions for pattern classification, was analyzed. The authors use a pre-trained ResNet-101 deep neural network model. The accuracy of classification of a set of images from a camera mounted on a finger reaches 92%. The work [4] describes not the application of neural networks but image processing algorithms. The authors propose two new descriptors: cCENTRIST and tCENTRIST. On the Clothing Attribute Dataset, the cCENTRIST method achieved an accuracy of 74.97%, while the highest accuracy of 84.23% was obtained on the Fashion Dataset.

The authors of the work [5] propose a model based on the AlexNet and VGG_S architectures, achieving an accuracy of 77.8% on the Clothing Attribute Dataset with 6 different classes and 84.5% on the Fashion Dataset with 5 texture classes.

The structure of the input data set. The following patterns are defined for classification: “checkered”, “dotted”, “vegetation/floral”, “print”, “solid”, and “striped”. For the task, 100 images were selected manually for each class, for a total of 600.

The architecture of the applied neural network. Figure 1 shows the architecture of the convolutional neural network.

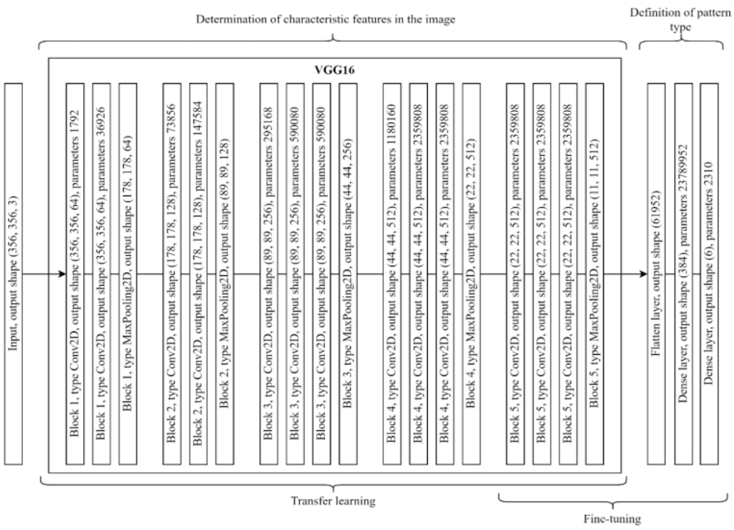


Fig. 1. Network architecture for pattern classification of clothing elements

Conducted experiments. The dataset was divided according to the following distribution: training – 80%, validation – 10%, test – 10%. The batch size was set to 32. The input image size is 356x356.

First, a manual selection of hyperparameters was carried out to select the model with which the research would be carried out. VGG16, VGG19, InceptionV3, Xception, and ResNet50 models were used for testing. For each model, the weights on the ImageNet dataset were loaded, the fully connected layers were discarded, and training was disabled for all network elements. A sequential network was created, with the VGG16, VGG19, InceptionV3, Xception, and ResNet50 networks as the first element, followed by a smoothing layer, a fully connected layer with 256 neurons, adding the ReLU activation function and a dropout with value 0.5, and a fully connected layer of 6 neurons with the SoftMax activation function. The loss function was "categorical cross-entropy", and the Adam optimizer was used with a learning rate of 0.001. Further research was conducted by testing the VGG16 model, which showed the highest accuracy.

It was at this stage that Keras-Tuner was used. A Bayesian Optimization search algorithm was applied to Keras-Tuner. The VGG16 network weights were loaded on the ImageNet dataset, the fully connected layers were dropped for classification, and training was disabled for all network elements. A sequential network was created, where the first element is a part of the VGG16 network, then a flatten layer was added. The number of fully connected layers ranges from 1 to 2, with each layer having 32 to 512 neurons in increments of 32. The activation functions of fully connected layers that could be used are ReLU, Sigmoid, Tanh, ELU, and SELU. A dropout with a value of 0.5 was added for each fully connected layer. After that, a fully connected layer of 6 neurons was added and the SoftMax activation function was specified for it. The optimizers that could be used are Adam, RMSprop, and SGD. The metrics that were calculated are Accuracy, Precision, and Recall. Keras-Tuner performed 10 trainings of the network. The optimization was carried out by minimizing the value of the loss function on the validation set.

A model was built based on the set of hyperparameters with the lowest value of the loss function. The VGG16 network weights were loaded on the ImageNet dataset, the fully connected layers were dropped for classification, and training was disabled for all network elements. A sequential network was created, where the first element is a part of the VGG16 network. Then, a flatten layer, a fully connected layer containing 384 neurons with the Sigmoid activation function, a dropout with a value of 0.5, and a fully connected layer of 6 neurons with the SoftMax activation function were added. The loss function was "categorical cross-entropy", and the RMSprop optimizer was used with a learning rate of 0.001. There were 50 epochs.

At the next stage, fine-tuning was used, involving the "unfreezing" of the layers of the 5th block of the VGG16 neural network. The work was carried out with the trained model, the hyperparameters for which were obtained from Keras-Tuner. The trained model was loaded into memory. For the layer containing the convolutional part of the VGG16 model, the cycle that "unfreezes" the layers of the 5th block was described. The loss function was "categorical cross-entropy", and the RMSprop optimizer was used with a learning rate of 0.00001. There were 50 epochs. Table 1 shows the metrics of the obtained model.

Table 1. Metrics of the model with "unfreezing" of the layers of the 5th block of the convolutional part of the VGG16 layer

Sample	Loss	Accuracy	Precision	Recall
Training	0,0404	0,9999	0,9999	0,9999
Validation	0,1000	0,9999	0,9999	0,9688
Test	0,2078	0,9500	0,9655	0,9333

Figure 2 shows graphs of accuracy and loss functions for the model with "unfreezing" of the layers of the 5th block of the convolutional part of the VGG16 layer.

The trained network was tested on new images. Figure 3 shows the classification results for the "dotted" class (accuracy "99.68%") and the "floral" class (accuracy "98.62%").

Limitations. Frozen layers of the VGG16 network limit its adaptability for clothing pattern recognition. Fine-tuning has been partially applied, but further unfrozen layers, starting from the top blocks, could enhance learning from specific data. The study uses large images (356x356), which may require significant memory and processing resources. A solution to this is to reduce the image size, for example, to 224x224. The model classifies 6 types of patterns, which is insufficient for real-world applications. A solution is to expand the number of classes.

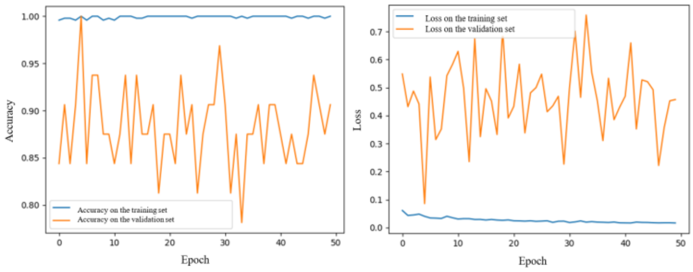


Fig. 2. Plots of accuracy and error functions for the model with "unfreezing" of the layers of the 5th block of the convolutional part of the VGG16 layer

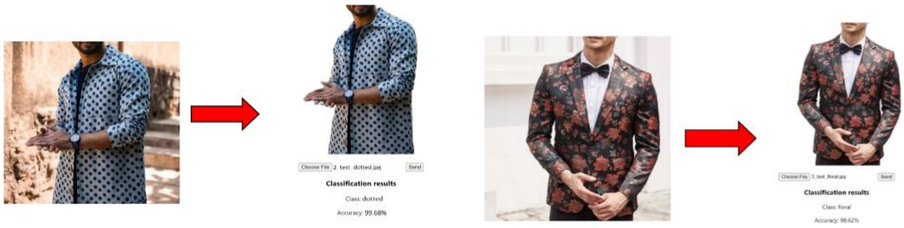


Fig. 3. Testing the user interface and the trained model by sending an image with dotted and floral patterns

Discussion of the obtained results. Using the transfer learning method and the fine-tuning approach, the final version of the network with a training accuracy of 99.99%, a validation accuracy of 99.99%, and a testing accuracy of 95% was obtained.

The scientific novelty of the obtained research results is that, based on machine learning, with the use of transfer learning methods and fine-tuning approaches, the pattern classification model of clothing elements was further developed. The practical significance of the research results lies in the application of means for the classification of patterns of clothing elements using machine learning.

Conclusions. Models and tools for increasing the efficiency of the clothing element pattern classification using machine learning were researched and implemented. The study confirmed that for the developed model, an increase in the efficiency of the clothing element pattern classification was achieved.

1. Venkatkumar, R. (2022, December 3). Background Removal (General (Pixel-wise) vs Deep learning) Approach. *Medium*. Retrieved from https://medium.com/@VK_Venkatkumar/background-removal-general-pixel-wise-vs-deep-learning-approach-4f919fe3dd3a.
2. Teslyuk, V. M., & Ivasiv, S. S. (2023). System for recognizing clothing items and their colors in an image. *Ukrainian Journal of Information Technology*, *5*(2), 25–32. https://doi.org/10.23939/ujit2023.02.025.
3. Stearns, L., Findlater, L., & Froehlich, J. E. (2018). Applying Transfer Learning to Recognize Clothing Patterns Using a Finger-Mounted Camera. In *Proceedings of the 20th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '18)* (pp. 349–351). Association for Computing Machinery. https://doi.org/10.1145/3234695.3241015.
4. Dey, E., Tawhid, M. N. A., & Shoyaib, M. (2015). An Automated System for Garment Texture Design Class Identification. *Computers*, *4*, 265-282. https://doi.org/10.3390/computers4030265.
5. Islam, S. S., Dey, E. K., Tawhid, M. N. A., & Hossain, B. M. M. (2017). A CNN Based Approach for Garments Texture Design Classification. *Advances in Technology Innovation*, *2*(4), 119–125. Retrieved from https://ojs.imeti.org/index.php/AITI/article/view/366.

Kateryna Tkachuk[1], Nataliia Kustra[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, kateryna.i.tkachuk@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, nataliia.o.kustra@lpnu.ua

INTEGRATING CLOUD-EDGE COMPUTING FOR NEURAL NETWORKS IN MOBILE SMART SYSTEMS

Introduction. In the modern world, mobile smart systems, such as smartphones, tablets, smartwatches, and various IoT devices, play a key role in our daily lives by performing complex tasks that require significant computational resources. The growing demand for integrating artificial intelligence technologies into these devices increases the need for neural networks to handle tasks like image recognition, natural language processing, prediction, and personalization of user experience. However, the limited computational resources of mobile devices pose significant challenges for deploying complex deep learning models that require substantial computing power. To solve this problem, a promising approach is the integration of cloud-edge computing, which allows for the optimal distribution of computing tasks between the cloud and edge devices.

Materials and methods. This approach provides an effective balance between performance, energy efficiency, and data transmission delays, allowing mobile smart systems to consistently perform intelligent functions with minimal delays and optimal resource utilization. This ensures high system performance, fast response times, and energy efficiency, contributing to the efficient operation of smart functions on mobile devices. The main advantages are:

1. Reduce latency and increase productivity. Integrating edge computing reduces the distance that data has to travel before being processed, which reduces latency and increases the speed of information processing. This is critical for real-time applications such as autonomous driving, augmented reality, and pattern recognition. For example, a study showed that edge computing can reduce video processing latency by 50-60% compared to traditional cloud computing [1].
2. Energy efficiency. By processing data locally on edge devices, the need to transfer large amounts of information to the cloud is reduced, which reduces the power consumption of mobile devices. This is especially important for battery-powered devices such as smartphones, where energy efficiency is a critical parameter. Studies show that the use of edge computing can reduce energy consumption by up to 30% compared to centralized processing in the cloud [2].
3. Scalability and flexibility. Cloud-edge computing offers a flexible system where computational resources are dynamically scaled based on demand, allowing seamless expansion or contraction of processing capabilities without requiring significant investment in additional physical infrastructure. During periods of high demand, edge devices can manage increased local data processing, while cloud resources can take over when these needs subside, maintaining overall system efficiency. This scalability is particularly beneficial for complex applications like real-time data analysis or AI-driven systems, preventing performance bottlenecks.

The cloud-edge architecture also enables real-time adjustments to computing capacity, which is crucial for applications like autonomous vehicles or industrial automation, where data processing requirements can shift unpredictably. This flexibility allows developers to adjust system resources in response to varying workloads, ensuring high performance even in demanding or rapidly changing environments. Moreover, it reduces the need for on-site hardware upgrades, as edge devices can offload tasks to the cloud, optimizing resource allocation without sacrificing performance.

4. Improved data security and privacy. Edge computing improves data security by reducing the necessity to transfer sensitive information to centralized cloud servers, minimizing potential vulnerabilities such as interception or unauthorized access. In industries such as healthcare and finance, where data privacy is critical, local data processing ensures sensitive information remains on the device, avoiding exposure during transmission. By processing personal data locally, edge computing also supports compliance with privacy regulations like GDPR and HIPAA, which mandate strict data localization. Real-time encryption and device-level security measures further safeguard the data against breaches, offering robust protection in real-time environments.

In addition, user-specific data such as personal preferences, medical information, and biometric data remain with the device or local network, further enhancing privacy and reducing the need to go through external servers that could be hacked. This localized processing not only ensures that raw data is securely contained but also maintains a higher degree of privacy by minimizing third-party access to sensitive information. Devices implement real-time security measures, such as encryption and authentication protocols, to provide an additional layer of protection, ensuring that data is transmitted and stored securely without the risk of exposure.

5. Reduced network load. Processing data at the edge significantly reduces the volume of data transmitted to cloud servers, easing the strain on network infrastructure. By handling data locally, the load on communication channels is minimized, decreasing bandwidth usage and preventing network congestion—an essential consideration in IoT environments where countless devices continuously generate data. In time-sensitive applications like autonomous driving or real-time analytics, the reduction in network load enhances responsiveness and boosts system efficiency by lowering latency.

Edge computing also enables the filtering, preprocessing, or summarizing of large datasets before forwarding them to the cloud. This approach not only optimizes network traffic but also enhances processing efficiency by ensuring that only the most relevant data is transmitted for further analysis or long-term storage. This reduces the strain on cloud systems and supports scalability for larger IoT networks, allowing for seamless integration of more devices without overwhelming the infrastructure. This division of labor between the edge and cloud ensures that the system remains capable of handling growing data volumes while maintaining high performance and reduced operational costs.

Results. By distributing tasks between the cloud and edge devices, this approach optimizes performance, energy efficiency, and data security, all while minimizing latency and network load. The benefits of this hybrid architecture include reduced latency, increased energy efficiency, scalability, improved security, and reduced network load, which are crucial for the effective implementation of neural networks in mobile devices. This enables complex AI tasks like image recognition and natural language processing to be performed efficiently without sacrificing performance.

Conclusions. The integration of cloud-edge computing in mobile smart systems provides a powerful solution to the challenges posed by limited computational resources. Future advancements in cloud-edge computing will further enhance these systems' scalability, flexibility, and privacy, fostering smarter, more efficient mobile technology. However, ongoing research is needed to ensure that cloud-edge computing reaches its full potential. This research should focus on optimizing resource management, improving load balancing strategies between cloud and edge, and ensuring the reliability and robustness of distributed systems. In addition, exploring new ways to handle peak computing loads without compromising performance is critical to maintaining the high efficiency and effectiveness of cloud-edge computing in an increasingly AI-driven mobile ecosystem. As this technology matures, it will foster smarter, more efficient, and secure mobile devices that will contribute significantly to the evolution of mobile computing and AI applications.

1. Yu, W., Liang, F., He, X., Hatcher, W. G., Lu, C., Lin, J., & Yang, X. (2018). A survey on the edge computing for the Internet of Things. *IEEE Access*, *6*, 6900–6919. https://doi.org/10.1109/ACCESS.2017.2778504.
2. Zhou, Z., Chen, X., Li, E., Zeng, L., Luo, K., & Zhang, J. (2019). Edge intelligence: Paving the last mile of artificial intelligence with edge computing. *Proceedings of the IEEE*, *107*(8), 1738–1762. https://doi.org/10.1109/JPROC.2019.2918951.
3. Mukherjee, M., Matam, R., Mavromoustakis, C. X., Jiang, H., Mastorakis, G., & Guo, M. (2020). Intelligent edge computing: Security and privacy challenges. *IEEE Communications Magazine*, *58*(9), 26-31. https://doi.org/10.1109/MCOM.001.2000297.
4. Alnoman, A., Sharma, S. K., Ejaz, W., & Anpalagan, A. (2019). Emerging edge computing technologies for distributed IoT systems. *IEEE Network*, *33*(6), 140-147. https://doi.org/10.1109/MNET.2019.1800543.

Vladyslav Todorchuk[1], Iryna Kazymyra[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, vladyslav.todorchuk.mknus.2023@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, iryna.y.kazymyra@lpnu.ua

MODEL AND TOOLS FOR PREDICTING HEALTH STATUS CHANGES IN USERS WITH DIABETES BASED ON THEIR ACTIVITIES

Introduction. In recent years, technology has significantly advanced in improving health monitoring, especially for chronic diseases like diabetes. Diabetes, a condition affecting millions worldwide, requires continuous attention to diet, physical activity, and medication. Smart devices, such as wearables and sensors, have revolutionized health management by offering real-time data that can be used to predict potential health risks [1-3].

The proposed system is a mobile application designed to monitor and predict health risks based on a user's activities and insulin levels. By integrating data from Apple Watch, insulin sensors, and user manuals data inputs the system analyzes how different activities impact the user's health and predicts potential risks using Artificial Intelligence (AI).

The main goal is to empower users with diabetes to lead a worry-free life, with the app serving as a personal health assistant that provides timely warnings, preventing health crises before they occur as a result of user activities.

Methods and Materials.

The mobile app gathers data from two primary sources:

- Apple Watch, which tracks physical activity, heart rate, calories burned, and steps.
- Manual Data, which tracks activity like food or those that have not been tracked by other devices.
- Insulin sensor, which monitors blood glucose levels in real time.

These data streams are fed into a machine learning algorithm based on LLaMA 3.1, a large language model designed for complex pattern recognition tasks. The model was fine-tuned using a dataset from Indonesian users with diabetes, incorporating their medical and activity data. This specific dataset allowed the AI model to learn from a diverse population of diabetes patients, providing insights into how various activities affect blood sugar levels.

The AI model was trained using supervised learning techniques, where known outcomes (e.g., hypoglycemia or stable glucose levels) were mapped to activity patterns. This approach enabled the system to predict how different activities – such as walking, running, food, or resting – might affect individual users based on their health profiles [4].

Table 1. Models' comparison

Model version	Accuracy	Comments
Before Training	65%	Initial rule-based system, a higher rate of false positives and negatives due to a lack of personalized data.
After Training	90%	Significant improvement after fine-tuning the Indonesian medical dataset. The model became highly personalized and could better predict health risks for each user.

The following table shows the model's accuracy before and after fine-tuning with the Indonesian medical dataset.

By integrating user data from the Apple Watch and insulin sensors, and leveraging the fine-tuned LLaMA 3.1 model, the system can predict potential health risks in real-time. It provides early warnings to users, helping them adjust their activities or insulin intake accordingly, which ultimately improves their quality of life [5-6].

Results. The research on integrating AI into the mobile application for diabetes management reveals significant practical implications for user health monitoring. While the system is still in development, initial tests demonstrate promising capabilities of the AI model to predict potential health risks based on user activities. The application aims to achieve a rapid response time, ensuring that users receive timely warnings, which is crucial for managing their health effectively.

Although the system has not yet reached full optimization, the integration of AI is designed to deliver low-latency performance, making it conducive to real-time applications. The app operates efficiently to process user data from activities tracked via the Apple Watch and insulin sensors, allowing individuals with diabetes to make informed decisions regarding their daily activities and insulin management. By providing timely insights and alerts, the app serves as a valuable tool for enhancing user safety and empowering users to lead healthier lives with greater peace of mind. This capability not only improves health outcomes but also fosters a sense of security among users, reducing anxiety related to diabetes management.

Conclusions. This research has successfully advanced the field of diabetes management by developing and evaluating a real-time system that leverages AI for accurate predictions of health changes based on user activities. The primary contributions of this work include the adaptation and training of the LLaMA 3.1 model on a dataset of Indonesian medical data, which significantly improved predictive accuracy. By providing users with timely warnings, the system aims to empower individuals with diabetes to lead healthier, more peaceful lives.

1. Diabetes self-management education (DSME) methods using web-based and application. (n.d.). *ResearchGate*. https://www.researchgate.net/publication/375683258_Diabetes_Self-Management _Education_DSME_Methods_using_Web-Based_and_Application
2. Fitzgerald, J. E., & Akbari, A. (2020). Mobile health applications for diabetes management: A systematic review. *Journal of Diabetes Science and Technology, 14*(2), 393-402. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3840330/
3. Smilianets, F. A., & Finogenov, O. D. (2024). Review of disease identification methods based on computed tomography imagery. Ukrainian Journal of Information Tecnology, 6(1), 95–100. https://doi.org/10.23939/ujit2024.01.095
4. A review of whole-medical systems and holistic care approach for type 2 diabetes and associated metabolic syndrome. (n.d.). *ResearchGate*.
5. Top ten tips for successfully implementing a diabetes telehealth program. (n.d.). *ResearchGate*. https://www.researchgate.net/publication/340043233_Top_Ten_Tips_for_Successfully_Implementing_a_Diabetes_Telehealth_Program
6. Zhang, Y., & Xu, Y. (2021). The role of artificial intelligence in diabetes management. *International Journal of Diabetes Research, 10*(1), 1-8. https://www.researchgate.net/publication/282220429_A_Comparison_of_Supervised_Machine_Learning_Techniques_for_Predicting_Short-Term_In-Hospital_Length_of_Stay_among_Diabetic_Patients

Oksana Troian
[1]*Lviv Polytechnic National University, Lviv, Ukraine, oksana.a.troian@lpnu.ua*

ARTIFICIAL INTELLIGENCE-BASED TESTING OF MEDICAL SYSTEMS

Introduction. Digital technology is driving rapid, fundamental changes to the delivery of healthcare — from streamlining the detection, diagnosis, and treatment of NCDs to empowering clinicians. For example, by creating algorithms from data sets that reflect diverse populations, AI can help reduce the bias that is too often present in healthcare ecosystems, creating barriers to access. Healthcare technology can reach more people in new ways and expand access to care for people in different communities. Using innovation to reach more people in more places will help address global healthcare imbalances rather than relying just on healthcare technology. Over 76 million people's lives were made better in 2022 by Medtronic company [1]. Committed to enhancing outcomes, from AI and personalization to community-based connected care.

Citrine medicine has emerged as a result of the digitization of the major spheres of human activity. The main goals of digital medicine are illness prevention and maintaining the required standard of living for humans. Based on this, the following fundamental directions of digital medical development may be recognized in terms of invention and development at the current level of artificial intelligence (AI) technologies development:

1. mobile medical monitors that use smart sensors to measure the primary medical parameters in real-time and integrate them into a wireless sensor network (WSN);
2. mobile primary processing tools that use smart sensors to gather the measured medical parameters;
3. mobile ways to send data from a medical monitor that characterizes a patient's state in real-time to distant decision-making centers;
4. wearable injectors that can be remotely managed to introduce certain medications (drugs) that sustain human life.

To fulfill these demands, over six reputable businesses are creating families of medical analog interfaces that wearables or body medical sensors can be connected to for real-time registration and primary processing of measured medical data [2]. The same specifications apply to all multi-parameter wearable medical monitors: they must be reliable, small, and able to operate for extended periods without requiring battery replacement or recharging [3].

Materials and methods. The purpose of the work: testing the principles of applying artificial intelligence technologies in medicine. Testing with artificial intelligence, in particular machine learning and neural techniques networks, is frequently used for the automation process and enhancing the performance of mistake detection. Neural networks and intelligent machine algorithms enable to automate testing, shorten the time needed to implement it and improve mistake detection efficiency.

Additionally, the application of artificial intelligence improves the precision of software behavior prediction under various operating settings. Thus, the study's relevance lies because researching artificial intelligence techniques for software testing is a crucial step toward improving the process' efficacy, cutting expenses, and guaranteeing high-quality software for medical applications.

Medical software testing is an increased responsibility for the result of a specialist's work. Information systems provide doctors with various tools that facilitate decision-making: calculating the dosage of drugs, and prescribing this or that therapy. If information security professionals identify common, obvious source scenarios, then the testing department should also verify the absence of critical errors in the code. It should be noted that personal and clinical data repositories are separated. Thus, access management and processing of the patient's personal information and clinical data are different processes. When testing, it is necessary to check the possibility of making a request from the database bypassing authorization.

The general scheme for processing information of the medical service platform is presented in Fig. 1 - remote medical patient monitor.

A special place in mobile medical diagnostics is occupied by applications that allow users to receive medical advice, keep a log of symptoms, remind them to take medication, and also use integrated sensors to collect medical data. Such applications can promote early detection and treatment of a variety of conditions. Therefore, it is very important to test such applications, and with the capabilities of Artificial Intelligence, this is becoming an increasingly simple method of testing.

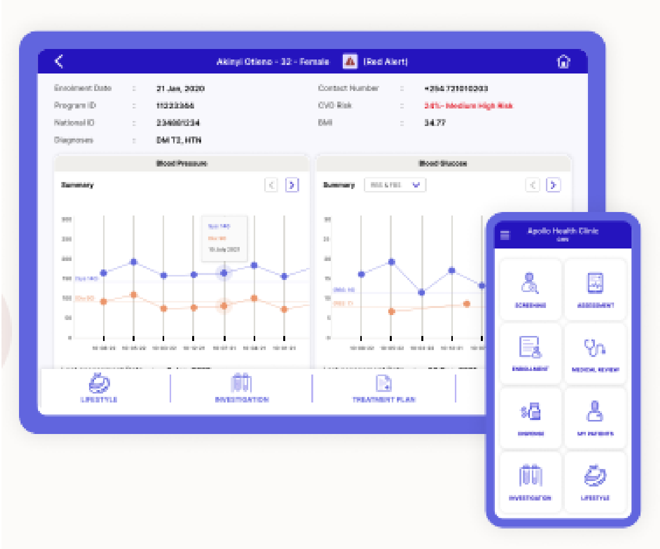


Fig. 1. Integrated Primary Care Managment

The international standard IEC 62304 is a norm that defines the requirements for the development life cycle of medical software and software for medical devices. It is agreed by the European Union (EU) and the United States (United States of America) and can therefore be used as a benchmark for regulatory compliance in both of these markets. This standard applies to the development and maintenance of software for medical devices. Testing enables to verification of functionality and performance in addition to locating and removing potential faults. Benefits and drawbacks of this strategy include:

1. Automation testing of speed. The ability of artificial intelligence systems to swiftly evaluate vast amounts of data facilitates the easier and quicker determination of test findings.
2. Testing precision. More accurate findings are possible because algorithms can identify and fix possible mistakes.
3. Testing tailored examination. With machine learning, you may build customized models that consider the particular qualities of each user.
4. Testing challenges related to security and privacy. Applying AI integration is not without security risks. maintaining the security of information processing and maintaining the confidentiality of medical data.

The proposed model of using artificial intelligence methods for software testing is focused on external repositories, but this approach can be easily adapted to any system. Test collection is currently done in a sequential manner, which, although simple, can be time-consuming, especially for large open-source projects, for which the proposed software model may be most useful. The goals of the AI portfolio are to enhance patient outcomes and/or boost the effectiveness of the healthcare system. By using machine learning models, for instance, we can forecast a patient's propensity to visit the institution following a referral and to continue receiving care after enrollment. This enables health systems to provide more effective support to patients who might need it. The medical testing system is a sophisticated collection of modules, each of which carries out a distinct function in the planning, arranging, and assessment of patient data. The way these modules work together to build a cohesive system offers accuracy, flexibility, and efficiency when assembling a patient history and diagnosing a disease.

That is why it is critical to use unit testing, which focuses on validating the proper operation of individual program components or modules:

1. Testing user registration module simplifies entering new users into the system, allowing them to access analyses, conclusions, and treatments.
2. The testing question selection module uses an automated solution for selecting questions based on defined parameters, thereby ensuring the creation of individualized tests that correspond to health indicators and difficulty levels.
3. In the administration module, we can test an interface for managing the platform, including the ability to configure access rights, create users, and monitor and control system parameters.
4. The integration module with third-party systems tests wide opportunities for data exchange with other platforms, thus expanding the functionality of data and conclusions of various doctors.

The test results showed that the AI-based demonstrates a high level of accuracy and efficiency, indicating its significant potential to improve the testing process in medical systems. The introduction of asynchronous requests to artificial intelligence made it possible to increase the performance of the platform, ensuring its stable operation even under high load. Load testing has shown that the platform can efficiently handle a large number of simultaneous requests, making it reliable and safe for both patients and users. Automating the creation of tests with the use of artificial intelligence promotes efficiency, accuracy, and personalization of data, responding to modern medical challenges. Fig. 2 shows a screenshot that demonstrates examples of automation testing results with artificial intelligence.

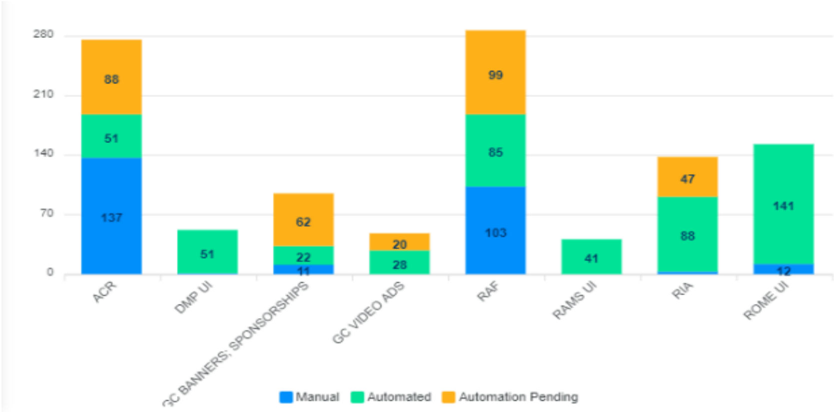


Fig. 2. Testing results

The work examined methods for using technologies to predict test results for open-source software projects, came to conclusions about the viability of using AI, tested medical systems using AI techniques, and identified many ways to significantly increase testing quality by leveraging AI to obtain potential research results with test case locations predicted.

Conclusions. The further development of mobile information technology in medicine shows significant potential for improving the accuracy and visibility of diagnostics. Integrating artificial intelligence into test score research and using machine learning for analysis open new perspectives in the field of medical research. An important aspect is the need for further research and improvement of algorithms to ensure high efficiency and reliability of the system. In addition to technical issues, it is also important to consider the ethical and sociocultural aspects of introducing such technologies into medical practice.

1. Medtronic Company. [Electronic resource]: https://europe.medtronic.com/xd-en/our-impact/healthcare-access-technology-equity.html
2. Broeders J.-H. Wearable Market Welcomes Biomedical All-Rounder. www.analog.com.
3. Romanov, V., Galelyuka, I., Voronenko, O., Kovyrova, O., Dzyadevych, S., Shkotova, L. (2019). Smart Sensors and Computer Devices for Agriculture, Food Production Process Control and Medicine. ICCTA 2019, 29-31 October, Alexandria, Egypt. 9-12.

Volodymyr Tsapiv[1], Iryna Kazymyra[2]
[1] Lviv Polytechnic National University, Lviv, Ukraine, volodymyr.tsapiv.mknus.2023@lpnu.ua
[2] Lviv Polytechnic National University, Lviv, Ukraine, iryna.y.kazymyra@lpnu.ua

GESTURE CONTROL IN REAL-TIME: A FRAMEWORK FOR 3D HAND POSE RECOGNITION FOR SMART DEVICES AND TOUCHLESS INTERFACES

Relevance of the research. Hand gestures and gesticulations are a common form of human communication. It is therefore natural for humans to use this form of communication to interact with machines as well. Dynamic hand gesture recognition in real-time has become increasingly important in the fields of human-computer interaction (HCI), virtual reality (VR), augmented reality (AR), and robotics. Traditional input methods like keyboards, mice, and touchscreens are often limiting or impractical in certain environments, such as VR/AR applications, where hands-free interaction is preferable.

The practical applications of real-time dynamic hand gesture recognition are diverse and impactful across various domains [1]. In smart environments, such as smart homes, smart TVs, and smart cars, hand gesture control offers a more convenient and hands-free method of interaction, enhancing user experience and accessibility. In public spaces, touchless interfaces for ATMs, kiosks, and elevators can significantly reduce the spread of infections by minimizing physical contact with shared surfaces, addressing a critical public health concern.

In healthcare settings, this technology allows surgeons and medical personnel to interact with digital tools within sterile environments using hand gestures, thus maintaining strict hygiene standards without the need to touch any surfaces. Additionally, in the realm of AR/VR gaming, dynamic hand gesture recognition greatly enhances immersive experiences by enabling players to interact with virtual environments naturally and intuitively, eliminating the reliance on physical controllers.

Methods and materials. Many existing methods [2-3] in the literature effectively address offline classification problems, where the input data consists of pre-segmented sequences of joint positions. These approaches are typically well-suited for scenarios where all data is available beforehand, allowing for detailed analysis and classification. However, this offline approach does not apply to real-time scenarios where decisions must be made on the fly as data streams continuously. Furthermore, existing online classification methods [4], which are designed to operate in real-time, often require significant computational resources, making them impractical for CPU-only environments, particularly in applications where latency and efficiency are critical.

To address these challenges, this research proposes a novel approach that modifies a high-performance offline classification method to be suitable for online inference. The method is designed to work in conjunction with Google Mediapipe [5], a widely used framework for hand pose estimation, but it is not restricted to this platform. The core of the proposed method is based on the DD-Net [6] architecture, a 1D convolutional neural network (CNN) designed to process sequential data efficiently.

The model takes as input three distinct features extracted from the hand pose data:

1. Slow Motion: The difference in 3D joint positions between two consecutive frames (frames n and n+1).
2. Fast Motion: The difference in 3D joint positions between two non-consecutive frames (frames n and n+2).
3. Location-Viewpoint Invariant Feature: A triangular matrix representing the distances between all joints, which remains invariant to changes in viewpoint.

These three input features are individually processed through 1D convolutional blocks, which project them into an embedded space. The embeddings generated from each feature are then combined and fed into a series of 1D convolutional and fully connected (FC) layers, which ultimately predict the gesture class.

Inspired by the architecture proposed in [6], modifications were made to enhance the model's performance and robustness. Specifically, the model was extended to include two prediction heads. The first head predicts the gesture class from a predefined set of gestures, including a non-gesture class. The second head is an auxiliary task, predicting one of three broader categories: static, dynamic, and non-gesture. This auxiliary task reinforces the model's ability to distinguish between gestures and non-gestures, thereby improving the overall accuracy of the system. Additionally, the convolutional blocks in the network were updated with residual connections, similar to those used in ResNet [7] architectures, to facilitate deeper networks and improve gradient flow during training.

During inference on streaming video data, the model processes a sliding window of 16 frames, corresponding to approximately 0.54 seconds of temporal context. The first prediction head is used to make real-time predictions, while the second head enhances the detection of non-gestures during training. To ensure stability and robustness in real-time prediction, a voting procedure is applied to the most recent predictions, which helps to smooth out transient errors and produce more reliable results.

Results of research. The research involved successfully training modified neural networks on the NVGesture [8] dataset, which comprises over 1,500 video sequences depicting 25 distinct hand gestures performed by 20 participants. The dataset includes both static gestures (such as "thumb up" and "ok") and dynamic gestures (such as "stop" and "right"). Given that the dataset was originally in the form of images, additional preprocessing was required. To address this, I developed an automatic annotation procedure utilizing Google's Mediapipe solution to prepare the data for training.

The evaluation of the trained networks was conducted in two phases. The first phase involved assessing the model's performance on the test partition of the NVGesture dataset. Standard metrics, including accuracy, recall, precision, and F1 score, were used to quantify the model's effectiveness. The second phase involved a real-time evaluation using a webcam to simulate practical application scenarios. For this manual real-time evaluation, each gesture type was attempted three times, and the success rate was recorded based on the number of successful attempts.

Both phases of evaluation were conducted on 3D data (where Mediapipe estimates hand poses in world coordinates) and on 2D keypoints (using normalized 2D coordinates of detected joints). The results from these evaluations provided valuable insights into the model's performance, both in controlled dataset conditions and in real-world application scenarios.

Table 1. Evaluation with metrics

	Accuracy	Recall	Precision	F1
2D data	0.794	0.794	0.763	0.768
3D data	0.784	0.758	0.784	0.751

Table 2. Manual evaluation

	Static	Dynamic	Non-gesture
2D data	0.345	0.567	0.790
3D data	0.893	0.712	0.921

Overall, by employing the 3D data approach, we managed to outperform the results reported in the original paper on the NVGesture dataset, which claimed an accuracy of 0.74 using only color data. This demonstrates the effectiveness of our approach and its potential for more accurate and reliable real-time hand gesture recognition.

Practical value. The research on dynamic hand gesture recognition in real time has significant practical implications across various fields, particularly given the promising performance of the developed classification network. Despite not yet being fully optimized, the network achieves an average inference time of 5 ms, which is notably fast and conducive to real-time applications. When combined with Google Mediapipe's hand pose detection, which runs at 22 ms per inference, the overall system remains within the threshold required for responsive and seamless user interactions.

This low-latency performance is particularly advantageous in applications where quick and accurate hand gesture recognition is critical. For instance, in smart home or smart car systems, the ability to process gestures in real-time ensures that user commands are executed immediately, enhancing user experience and safety. In public spaces, where touchless interfaces are increasingly important for health reasons, the system's rapid response time can provide smooth and efficient interaction, reducing user frustration and improving accessibility.

Conclusions. This research has successfully advanced the field of dynamic hand gesture recognition by developing and evaluating a real-time system that leverages both 3D and 2D data for accurate and efficient gesture classification. The primary contributions of this work include the adaptation and training of modified neural networks on the NVGesture dataset, the implementation of an automatic annotation procedure using Google Mediapipe, and the comprehensive evaluation of the system's performance in both controlled and real-time environments.

1. Chornenkyi, V. Ya., & Kazymyra, I. Ya. (2023). Research of the models for sign gesture recognition using 3D convolutional neural networks and visual transformers. Ukrainian Journal of Information Technology, 5(2), 33–40. https://doi.org/10.23939/ujit2023.02.033
2. Emporio, M., Caputo, A., Giachetti, A., Cristani, M., Borghi, G., D'Eusanio, A., Le, M., Nguyen, H., Tran, M., Ambellan, F., Hanik, M., Nava-Yazdani, E., & Von Tycowicz, C. (2022). SHREC 2022 track on online detection of heterogeneous gestures. Computers & Graphics, 107, 241–251. https://doi.org/10.1016/j.cag.2022.07.015
3. Köpüklü, O., Gunduz, A., Kose, N., & Rigoll, G. (2020). Online Dynamic Hand Gesture Recognition Including Efficiency Analysis. In IEEE Transactions on Biometrics, Behavior, and Identity Science, 2, 85–97. https://doi.org/10.1109/tbiom.2020.2968216
4. Caputo, A., Giachetti, A., Soso, S., Pintani, D., D'Eusanio, A., Pini, S., Borghi, G., Simoni, A., Vezzani, R., Cucchiara, R., Ranieri, A., Giannini, F., Lupinetti, K., Monti, M., Maghoumi, M., LaViola, J. J., Jr, Le, M., Nguyen, H., & Tran, M. (2021). SHREC 2021: Skeleton-based hand gesture recognition in the wild. Computers & Graphics, 99, 201–211. https://doi.org/10.1016/j.cag.2021.07.007
5. Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C., & Grundmann, M. (2020). MediaPipe Hands: On-device real-time hand tracking. arXiv.org. https://arxiv.org/abs/2006.10214
6. Yang, F., Sakti, S., Wu, Y., & Nakamura, S. (2019). Make skeleton-based action recognition model smaller, faster and better. arXiv.org. https://arxiv.org/abs/1907.09658
7. He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition. arXiv.org. https://arxiv.org/abs/1512.03385
8. Molchanov, P., Yang, X., Gupta, S., Kim, K., Tyree, S., & Kautz, J. (2016). Online Detection and Classification of Dynamic Hand Gestures with Recurrent 3D Convolutional Neural Networks. (n.d.). IEEE Conference Publication | IEEE Xplore, 4207-4215. https://doi.org/10.1109/cvpr.2016.456.

Liubomyr Vytvytskyi[1], Anna Gurynovych[2]
[1] Lviv Polytechnic National University, Lviv, Ukraine, liubomyr.vytvytskyi.pp.2023@lpnu.ua
[2] Lviv Polytechnic National University, Lviv, Ukraine, anna.s.hurynovych@lpnu.ua

INTEGRATING PHYGITAL SOLUTIONS IN SMART CITIES TO ENHANCE ACCESSIBILITY

Introduction. It is stated that the population of the Earth is above 8 billion now. According to the World Health Organization (WHO), 16% or 1,3 billion live with disabilities. Furthermore, more than half of the impaired people live in cities. The key to improving their quality of life is smart cities. According to International Business Machines Corporation (IBM), a smart city is an urban area where technology and data collection help improve the quality of life as well as the sustainability and efficiency of the city operations. Technologies are the heart of this city type. The question is what can be done to alleviate the living conditions of disabled people. The proposed solution is the Phygital concept.

Background. The phygital concept was coined by Chris Weil in 2007. According to the marketing dictionary of Monash University, it is defined as a bridge between the digital world and the physical world with the purpose of providing unique interactive experiences for the user [1]. The most famous example of the concept being implemented is the Pokémon Go. It is an app where people need to catch Pokémons using their smartphone's camera and GPS aiming to grow their collection. The game, created in 2016, still has about 90 million active users in 2024.

Case studies. One of the challenges in implementing the phygital experience is working with old buildings that require refurbishment to meet the needs of the disabled. A noteworthy case is the implementation of the phygital experience in an old infrastructure of the New York subway system, first operated in 1904. In 2020 the Transit Tech Lab launched a startup competition, where they chose 9 companies to improve accessibility. Among them was Okeenea Digital with its audio-based indoor navigation app Evelity. It is adapted for all types of disabilities and can easily build the best route regarding a person's special needs. Later, the underground was equipped with beacons to advance the accessibility even more. The field of interest of Okeenea Digital covers not only underground and public transport but also includes workplaces, universities, museums, and hotels.

Another example worth mentioning is Tactile Studio, which specializes in designing educational solutions enhanced by sensory experiences – touch, sound, and smell. The Tactile studio promotes the phygital concept through the idea of the interactive hybrid between physical (gestures) and digital (screens). It requires programming, modern electronics, 3D modeling, and involving the leading experts. One of their greatest projects was visualizing 16 stations, creating a tactile trail at the Pavillon de l'Horloge at the Louvre [2]. On the path the visitors can find the archaeological sights and identify the specific period to which it belongs, as well as the explanation of the decorative elements with the ability to touch the reproductions. In this case, the sensitive consoles with infrared sensors are placed in every room, allowing interaction by passing the hand over it to light up the corresponding room and get the information about it. The Tactile studio provides a comprehensive experience with audio devices and Braille text. Furthermore, the Louvre is equipped with video materials with sign language specifically for hearing-impaired people. The actors in the videos wear historical costumes to make the experience more immersive and entertaining.

Another point we would like to cover is the significant progress in creating applications for visually impaired people. TUAT corp. – a South Korean company that introduced artificial intelligence in their application Sullivan+ in order to recognize and vocalize objects from photos. It helps blind and low-vision individuals experience the phygital concept and understand what is around them. The implementation of artificial intelligence makes this app accessible all around the world, irrespective of a particular city or country. Furthermore, Sullivan+ makes it easier to get acquainted with a new person because of the built-in Face recognition that can identify the age and gender. Color recognition supports single-color mode, which describes what color is in the center of the photo, and full-color mode, which indicates what color covers a large part of the entire screen. Alongside that, the Light brightness function helps visually impaired people facing problems with understanding how bright it is around them. It can also describe the surroundings and thus help to navigate unfamiliar spaces [3].

Similarly, a Lazarillo application aims to make commuting easier and more efficient for disabled people. It offers audio navigation with real-time updates and multiple layers of information with the concept of the smart city in mind. As a person walks, Lazarillo will announce places of interest, streets, intersections, restaurants, shops, and transit areas. In addition, it can create routes within the map and guide the users on their way to the destination. The developers offer specific personalized services to particular businesses, which include not only mapping and creating the digital plan of the facilities but also adding accessible and interactive digital information to their physical space [4]. Applications like Lazarillo are easily applicable in smart cities as they do not require major refurbishment, but rather work with the existing environment and Big Data.

It is of utmost importance for Ukraine to implement the advancements focused on supporting people with disabilities, particularly in light of the full-scale Russian invasion. Furthermore, constant technological developments are to be considered, as they enable innovations to be combined with the existing surroundings. Ukraine promotes the phygital concept in smart cities too. For instance, in Lviv, there is a project that allows impaired people to do sightseeing and manage daily tasks. The extension to the Google Maps app is developing constantly and is available on any device with access to the Internet. It was introduced in February 2024. It includes more than 100 places for visitors with different types of disabilities, providing a more inclusive environment for tourists and citizens. Restaurants, hotels, museums, administrative buildings, libraries, hospitals, parks, and places for entertainment are listed, while the convenient search by category improves the user experience [5]. The Accessible City map is supported by the city council, making this project a state-endorsed one.

Conclusions. To sum up, a smart city is about improving the quality of life for every citizen using leading-edge technologies and data collection. The phygital concept is the cornerstone of the smart city. It allows the implementation of features for impaired people through interacting with the city using different devices. The examples covered in the paper allow us to trace the latest developments in the field. The first company Okeenea Digital developed an Evelity app that helps disabled people to find their way by installing Beacons in New York’s underground and improving the phygital experience. Secondly, Tactile Studio, which specializes in enhancing sensory experiences, has been cooperating with the Louvre Museum for 11 years, having visualized 16 stations with tactile path. Thirdly, TUAT corp. enriched the market by creating Sullivan+ which uses AI to process photos in real time, describing them using audio to facilitate commuting. Lazarillo creates the best route for its users and announces places of interest during a walk. A modern example of such technologies being implemented in Ukraine is the Accessible City map in Lviv. It provides a list of more than 100 accessible places. Projects like these create a phygital experience and have a significant impact on the lives of disabled people, making the reality of Smart cities closer. Thus, it is necessary to support and invest in projects to alleviate the everyday lives of impaired people.

1. Inclusive City Maker. (2022, May 24). How can a phygital experience improve the accessibility of your venue? Retrieved from https://www.inclusivecitymaker.com/phygital-experience-accessibility-venue/
2. Tactile Studio. (2022, January 4). The Louvre Museum’s Salle des Verres. Retrieved from https://tactilestudio.co/achievements/salle-des-verres-musee-louvre-architectural-model-sensory-device/
3. Sullivan Corp Blog. (2021, May 6). Tuat Corp. Retrieved from https://blog.mysullivan.org/
4. Lazarillo. (2024, September 4). Helping people with disabilities navigate the world. Retrieved from https://lazarillo.app/
5. Lviv travel. (2024, June 6). Map “Accessibile City”. Retrieved from https://lviv.travel/ua/news/karta-dostupne-misto-lviv

Rostyslav Zahvoiskyi[1], Iryna Kazymyra[2]
[1]Lviv Polytechnic National University, Lviv, Ukraine, rostyslav.y.zahvoiskyi@lpnu.ua
[2]Lviv Polytechnic National University, Lviv, Ukraine, iryna.y.kazymyra@lpnu.ua

AI TOOLS IN COMPLEX COMPUTING SYSTEMS MONITORING

Relevance of the research. Complex systems such as modern software architectures, ever-changing cloud infrastructures, or intelligent ecosystems are rooted in interplay between multiple systems and agents, which adapt over time. These failures can be caused by emergent behavior from parts of systems as they integrate or refine, the a priori inability to anticipate how these components will interact with other agents or people in total system performance over time [1]. It is paramount to monitor these deceptive developing subsystems to detect that anomalous conditions start appearing during their development, which could then lead to catastrophic failures. Yet, current monitoring methods fail to capture the dynamic nature of these systems and most often overlook people as active elements of the systems. Contemporary computing systems deliver advanced solutions for optimizing operations. For example, In reality, the behavior of these large, complex manufacturing systems is difficult to follow. One of the methods artificial intelligence (AI) provides is to improve both decision-making and monitoring [2].

This article presents an approach for capturing the dynamics of evolving properties within complex systems and detecting anomalies that might indicate impending system failures.

Research material. Detecting failures early in cloud-based software systems is essential for cutting down operational expenses, boosting service reliability, and improving user satisfaction [3]. Many current methods zero in on anomaly detection using metrics or a mix of metrics and log features. However, these techniques often end up being complex, hard to interpret, and tricky to implement and assess in real-world industrial settings [4]. These days, a lot of organizations are turning to cloud solutions like Amazon AWS CloudWatch, Microsoft Azure Monitor, Google Cloud Monitoring, and IBM Cloud Monitoring to tackle inefficiencies in their cloud infrastructures. There's a noticeable trend toward using tools specifically crafted to support monitoring in production environments. For this discussion, let's define Cloud Monitoring as the methods and tools we use to observe, review, and manage the operational workflow of cloud-based IT infrastructures. This involves both manual and automated techniques to ensure the availability and performance of cloud resources – think websites, containerized instances, applications, networks, and storage systems [5]. But beyond just these monitoring tools, weaving in artificial intelligence (AI) can seriously amp up cloud monitoring capabilities. AI stands to make monitoring processes more accurate and efficient by enabling smarter and more proactive management [4]. With technology constantly advancing and AI systems becoming more commonplace in industrial settings, this study zeroes in on creating an AI-based adaptive system for complex computing systems. The main aim is to design a system that can tell the difference between critical data that needs to be sent immediately and data that can be handled locally. This strategy is all about optimizing bandwidth and making sure we can respond quickly to any abnormal events. The big idea here is that by integrating AI directly into the monitoring system right where data is collected, we can cut down on network overhead while speeding up and sharpening the detection of critical events or patterns. Feedback is super important in the growth of any AI-based system. In our case, feedback is used for two main reasons: to check if the system's predictions are on point and to help the system adapt to new patterns and conditions [6,7]. By continuously integrating feedback, the system can evolve and get better over time. This means it doesn't just rely on the data it was initially trained on; it keeps adapting to changing conditions and improves its accuracy as it encounters new data. This ability to learn dynamically highlights the system's long-term effectiveness and adaptability in complex industrial environments. This study introduces an AI-based adaptive system designed for monitoring computing systems. The system smartly picks out critical data that needs immediate attention from data that can be processed on-site, helping to optimize bandwidth and ensure quick reactions to any unusual events. Ongoing feedback lets the system validate its predictions and adjust to changing conditions, boosting both its performance and accuracy as time goes on. In the end, this approach shows how AI-enhanced monitoring can reduce network strain, speed up issue detection, and ensure long-term adaptability in complex industrial settings.

Conclusion. As modern computing systems become increasingly intricate, there's a pressing need for advanced monitoring techniques that can adapt to dynamic environments and address emerging issues promptly. While traditional monitoring methods have their merits, they often fail to detect subtle interactions between components and agents within complex systems. The integration of AI tools offers a transformative solution – providing intelligent, proactive monitoring that not only identifies anomalies but also predicts potential failures before they become critical. This study showcases the effectiveness of an AI-based adaptive system for monitoring complex computing infrastructures. The system optimizes bandwidth usage and ensures swift responses to abnormal events by intelligently distinguishing between essential and non-essential data. It continually enhances its performance through feedback and adaptation, a dynamic learning capability that is crucial for maintaining long-term system reliability and efficiency in industrial environments. In summary, AI-enhanced monitoring holds significant promise for reducing operational costs, improving system reliability, and enabling quicker, more accurate issue detection. As the technological landscape continues to evolve, the adaptability and scalability of AI systems will be essential for managing the increasing complexity of industrial and cloud-based infrastructures – ensuring they remain efficient, resilient, and responsive to the ever-changing demands of the digital age.

1. EL Moussa, N. (2024). Smart Quality Monitoring for Evolving Complex Systems. In I*CSE-Companion '24: 2024 IEEE/ACM 46th International Conference on Software Engineering: Companion Proceedings*. ACM. https://doi.org/10.1145/3639478.3639784
2. Urgo, M., Terkaj, W., & Simonetti, G. (2024a). Monitoring manufacturing systems using AI: A method based on a digital factory twin to train CNNs on synthetic data. *CIRP Journal of Manufacturing Science and Technology, 50*, 249–268. https://doi.org/10.1016/j.cirpj.2024.03.005
3. Seniv M. M. (2024). A method of automating the process of calculating reliability indicators of software systems and their components. Ukrainian Journal of Information Tecnology, 6(1), 01-08. https://doi.org/10.23939/ujit2024.01.01.001
4. Hrusto, A., Runeson, P., & Ohlsson, M. C. (2024). Autonomous Monitors for Detecting Failures Early and Reporting Interpretable Alerts in Cloud Operations. In *ICSE-SEIP '24: 46th International Conference on Software Engineering: Software Engineering in Practice*. ACM. https://doi.org/10.1145/3639477.3639712
5. Aitiddir, H., & Kerzazi, N. (2023). Cloud Infrastructure Monitoring Using Splunk: Expectations and Challenges. In *2023 14th International Conference on Intelligent Systems: Theories and Applications (SITA)*. IEEE. https://doi.org/10.1109/sita60746.2023.10373692
6. Villegas-Ch, W., García-Ortiz, J., & Sánchez-Viteri, S. (2024). Towards Intelligent Monitoring in IoT: AI Applications for Real-Time Analysis and Prediction. *IEEE Access*, 1. https://doi.org/10.1109/access.2024.3376707
7. Kumara, V., M Sharma, D., Samson Isaac, J., Saravanan, S., Suganthi, D., & Boopathi, S. (2023). An AI-Integrated Green Power Monitoring System. In *Convergence Strategies for Green Computing and Sustainable Development* (c. 218–244). IGI Global. https://doi.org/10.4018/979-8-3693-0338-2.ch013

Наукове видання

Proceedings
of the 1st International Scientific and Practical Conference
«COMPUTATIONAL INTELLIGENCE AND SMART SYSTEMS»
CISS-2024

Матеріали
1-ої Міжнародної науково-практичної конференції
«ОБЧИСЛЮВАЛЬНИЙ ІНТЕЛЕКТ І СМАРТ-СИСТЕМИ»

Режим доступу: https://science.lpnu.ua/ciss-2024

Відповідальні за випуск – Мар'яна СЕНЕТА та Віктор ЗВАРИЧ

Підписано до друку 30.10.2024 р.
Папір ксероксний.
Формат 64×90/16. Гарнітура «Times».
Умов.друк.арк. 8,24. Наклад 300 прим.
Друк ПП «АРАЛ»
м. Львів вул. Козельницька, 4.