

Міністерство освіти і науки України
Харківський національний університет імені В. Н. Каразіна

ВІСНИК

Харківського національного університету
імені В.Н. Каразіна

Серія

«Математичне моделювання.
Інформаційні технології.
Автоматизовані системи управління»

Випуск 68

Серія заснована 2003 р.

BULLETIN

of V.N. Karazin Kharkiv National University

Series

«Mathematical Modeling.
Information Technology.
Automated Control Systems»

Issue 68

First published in 2003

Харків
2025

Засновник журналу Харківський національний університет імені В. Н. Каразіна, Харків, Україна. Рік заснування 2003. Періодичність: 4 випуски на рік. <https://periodicals.karazin.ua/mia>

Статті містять дослідження у галузі математичного моделювання та обчислювальних методів, інформаційних технологій, захисту інформації. Висвітлюються нові математичні методи дослідження та керування фізичними, технічними та інформаційними процесами, дослідження з програмування та комп'ютерного моделювання в наукоємних технологіях.

Для викладачів, наукових працівників, аспірантів, працюючих у відповідних або суміжних напрямках.

Наказом Міністерства освіти і науки України від 17.03.2020 № 409 наукове фахове періодичне видання Вісник Харківського національного університету імені В.Н. Каразіна серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління» включено до Категорії «Б» Переліку наукових фахових видань України за наступними спеціальностями: 113 – Прикладна математика; 122 – Комп'ютерні науки та інформаційні технології; 123 – Комп'ютерна інженерія; 125 – Кібербезпека.

Затверджено до друку рішенням Вченої ради Харківського національного університету імені В. Н. Каразіна (протокол № 32 від 22.12.2025 р.)

Редакційна колегія:

Азаренков М.О. (гол. редактор),

д.ф.-м.н., академік НАН України, проф., заступник генерального директора з наукової роботи ННЦ "Харківський фізико-технічний інститут" НАН України

Жолткевич Г.М. (заст. гол. редактора), д.т.н., проф. ФМІ ХНУ імені В.Н. Каразіна

Лазурик В.Т. (заст. гол. редактора), д.ф.-м.н., проф., ФТІ ХНУ імені В.Н. Каразіна

Споров О.Є. (відповідальний секретар), к.ф.-м.н., доц. ННІ КН та ШІ ХНУ імені В.Н. Каразіна

Золотарьов В.О., д.ф.-м.н., проф., ФТІНТ імені Б.І. Веркіна НАН України

Куклін В.М., д.ф.-м.н., проф., ННІ КН та ШІ ХНУ імені В.Н. Каразіна

Мацевитий Ю.М., д.т.н., академік НАН України, проф., ННІ КФ та енергетики ХНУ імені В.Н. Каразіна

Рассомахін С. Г., д.т.н., доц., ФКН ІВТ ХНУ імені В.Н. Каразіна

Стервоєдов М.Г., к.т.н., доц., ФКН ІВТ ХНУ імені В.Н. Каразіна

Толстолузька О. Г. д.т.н., с.н.с., доц., ННІ КН та ШІ ХНУ імені В.Н. Каразіна

Ткачук М. В., д.т.н., проф., ННІ КН та ШІ ХНУ імені В.Н. Каразіна

Шейко Т.І., д.т.н., проф., ННІ КФ та енергетики ХНУ імені В.Н. Каразіна

Шматков С. І., д.т.н., проф., ННІ КН та ШІ ХНУ імені В.Н. Каразіна

Раскін Л.Г., д.т.н., проф., Національний технічний університет "ХПІ"

Стрельнікова О.О., д.т.н., проф. ННІ КФ та енергетики ХНУ імені В.Н. Каразіна

Соколов О.Ю., д.т.н., проф., кафедра прикладної інформатики, університет імені Миколая Коперника, м. Торунь (Польща)

Яковлев С. В., д. ф.-м. н., проф., заступник директора ННІ комп'ютерних наук та штучного інтелекту ХНУ імені В.Н. Каразіна

Prof. **Harald Richter**, Dr.-Ing., Dr. rer. nat. habil. Professor of Technical Informatics and Computer Systems, Institute of Informatics, Technical University of Clausthal, Germany

Prof. **Philippe Lahire**, Dr. habil., Professor of computer science, Dep. of C. S., University of Nice-Sophia Antipolis, France

Адреса редакційної колегії: 61022, м. Харків, майдан Свободи, 6, Харківський національний університет імені В. Н. Каразіна, к. 534.

Тел. +380 (57) 705-42-81, Email: journal-mia@karazin.ua.

Мова публікації: українська, англійська.

Статті пройшли внутрішнє та зовнішнє рецензування.

Ідентифікатор медіа у Реєстрі суб'єктів у сфері медіа: R30-04456
(Рішення № 1538 від 09.05.2024 р Національної ради України з питань телебачення і радіомовлення. Протокол № 15)

© Харківський національний університет імені В.Н. Каразіна, оформлення, 2025

*The founder of the Journal is V. N. Karazin Kharkiv National University, Kharkiv, Ukraine.
Year of foundation 2003. The journal is published four times a year.*

<https://periodicals.karazin.ua/mia>

The articles are present research in the field of mathematical modeling and computing methods, information technologies, information security. New mathematical methods of research and management of physical, technical and information processes, research on programming and computer modeling in science-intensive technologies are covered.

For teachers, researchers, graduate students working in relevant or related fields.

By the order of the Ministry of Education and Science of Ukraine from 17.03.2020 № 409 scientific professional periodical Bulletin of V.N. Karazin Kharkiv National University series "Mathematical modeling. Information Technologies. Automated control systems" is included in Category "B" of the List of scientific professional publications of Ukraine in the following specialties: 113 – Applied Mathematics, 122 – Computer Science and Information Technology; 123 – Computer engineering; 125 – Cybersecurity.

Approved for publication by the decision of the Academic Council of V.N. Karazin Kharkiv National University (Minutes № 32 of 22.12.2025).

Editorial Board:

Azarenkov M.O. (Chief Editor), Acad. Of the NAS of Ukraine, Dr. Sc., Prof., Acting Director General of the National Science Center "Kharkiv Institute of Physics and Technology" of the NAS of Ukraine

Zholtkevich G.M. (Deputy Editor), Dr. Sc, Prof. MCS V.N. Karazin Kharkiv National University

Lazurik V.T. (Deputy Editor), Dr. Sc, Prof. CSD HTI V.N. Karazin Kharkiv National University

Sporov O.E., (Executive Secretary), Ph.D. Assoc. Prof, ESI of CS and AI V.N. Karazin Kharkiv National University

Zolotarev V.A., Dr. Sc, Prof. B. Verkin Institute for Low Temperature Physics and Engineering of the National Academy of Sciences of Ukraine

Kuklin V.M., Dr. Sc, Prof. CSD HTI V.N. Karazin Kharkiv National University

Matsevity Yu.M., Acad. Of the NAS of Ukraine, Dr. Sc., Prof., ERI ComPhys and Energy V.N. Karazin Kharkiv National University

Rassomakhin S.G., Dr. Sc, Prof. CSD HTI V.N. Karazin Kharkiv National University

Styervoyedov N.G., Ph.D. Assoc. Prof, CSD HTI V.N. Karazin Kharkiv National University

Tolstoluzka O.G., Dr. Sc, Assoc. Prof. ESI of CS and AI V.N. Karazin Kharkiv National University

Tkachuk M.V., Dr. Sc, Prof. ESI of CS and AI V.N. Karazin Kharkiv National University

Sheyko T.I., Dr. Sc, Prof. ERI ComPhys and Energy V.N. Karazin Kharkiv National University

Shmatkov S.I., Dr. Sc, Prof. ESI of CS and AI V.N. Karazin Kharkiv National University

Yakovlev S.V., Dr. Sc, Academicians of the NAS of Ukraine, Prof., Deputy Director of the Institute of CS and AI, V.N. Karazin Kharkiv National University, Ukraine

Raskin L.G., Dr. Sc, Prof. National Technical University "Kharkiv Polytechnic institute"

Strelnikova E.A., Dr. Sc, Prof., ERI ComPhys and Energy V.N. Karazin Kharkiv National University

Sokolov O.Yu., Dr. Sc, Prof. Nicolaus Copernicus University, Torun, Poland

Prof. **Harald Richter**, Dr.-Ing., Dr. rer. nat. habil. Professor of Technical Informatics and Computer Systems, Institute of Informatics, Technical University of Clausthal, Germany

Prof. **Philippe Lahire**, Dr. habil., Professor of computer science, Dep. of C. S., University of Nice-Sophia Antipolis, France

Editorial Address: 61022, Kharkiv, Svobodi sq., 6, V.N. Karazin Kharkiv National University, r. 534.

Phone. +380 (57) 705-42-81, Email: journal-mia@karazin.ua.

Language of publication: Ukrainian, English.

The articles pass internal and external review.

Media identifier in the Register of the field of Media Entities: R30-04456
(Decision № 1538 dated May 9, 2024 of the National Council of Television and Radio Broadcasting of Ukraine, Protocol № 15)

© V.N. Karazin Kharkiv National University, 2025

ЗМІСТ

▪ Білак Ю. Ю.	6
Гібридна нейромережева модель CNN+TRANSFORMER для прогнозування спектральних властивостей багатозв'язних структур	
▪ Голінко І. М.	20
Розробка цифрового двійника теплоспоживання для кіберфізичних систем в умовах концептуальної невизначеності	
▪ Костюченко А. В., Петрашенко А. В.	30
Метод генерації опису програмного коду з використанням моделі штучного інтелекту	
▪ Кошкін А. О., Стрельнікова О. О.	43
Аналіз згину багатозв'язних анізотропних плит з пружними включеннями	
▪ Лупандін А. В., Мороз О. Ю.	53
Аналіз сучасних нейромережевих методів обробки візуальної інформації в системах навігації високошвидкісних ПМР	
▪ Омельченко І. В., Струков В. М.	62
Архітектура рефлексивної пам'яті для адаптивного планування ієрархічних LLM-агентів у віртуальних середовищах	
▪ Стецюренко І. С., Петрашенко А. В.	70
Застосування методів машинного навчання для детекції зловмисного програмного забезпечення в дампах оперативної пам'яті	
▪ Супрун А. С., Бакуменко Н. С.	77
Адаптивне управління контекстом у RAG-системах для персоналізованих AI-асистентів	
▪ Трусов М. А., Турута О. П., Узлов Д. Ю.	84
Аналіз ефективності бібліотеки Resemblyzer для короткокомандної голосової автентифікації	
▪ Тюрдьо І. М., Кізілова Н. М.	98
Використання методів машинного навчання в сучасній математичній онкології	
▪ Шабельник Т. В., Прокопович С. В.	113
Експертна система оцінювання закономірностей мови методами непараметричної статистики	

CONTENTS

▪ Bilak Yu.	6
Hybrid Neural Network Model Based on CNN+Transformer for Predicting the Spectral Properties of Multilayer Structures	
▪ Golinko I.	20
Development of heat consumption digital twin for cyber-physical systems under conceptual uncertainty	
▪ Kostiuchenko A., Petrashenko A.	30
Method for generating source code description using an artificial intelligence model	
▪ Koshkin A., Strelnikova O.	43
Bending analysis of multiply-connected anisotropic plates with elastic inclusions	
▪ Lupandin A., Moroz O.	53
Analysis of Modern Neural Network Methods for Visual Information Processing in UAV Navigation Systems	
▪ Omelchenko I., Strukov V.	62
Reflective memory architecture for adaptive planning in hierarchical LLM agents in virtual environments	
▪ Stetsiurenko I., Petrashenko A.	70
Method of Power Supply Optimization for IoT Climate Monitoring System Based on Adaptive Algorithms	
▪ Suprun A., Bakumenko N.	77
Adaptive context management in RAG systems for personalized AI assistants	
▪ Trusov M., Turuta O., Uzlov D.	84
Analysis of the effectiveness of the Resemblyzer library for short-command voice authentication	
▪ Tiurdo I., Kizilova N.	98
The use of machine learning methods in modern mathematical oncology	
▪ Shabelnyk T., Prokopovych S.	113
An expert system for evaluating language patterns using nonparametric statistics	

УДК (UDC) 004.8:535.32:544.272

**Білак
Юрій Юрійович**

канд. фіз.-мат. наук, доцент, зав. кафедри програмного забезпечення систем ДВНЗ «Ужгородський національний університет», м. Ужгород, пл. Народна, 3, Україна, 88000
e-mail: yuriy.bilak@uzhnu.edu.ua
<https://orcid.org/0000-0001-5989-1643>

Гібридна нейромережева модель CNN+TRANSFORMER для прогнозування спектральних властивостей багатошарових структур

Актуальність. Прогнозування спектральних характеристик багатошарових матеріалів є ключовим завданням у фотоніці, оптоелектроніці та матеріалознавстві, оскільки точність моделювання визначає ефективність технологічних процесів і якість функціональних покриттів. Класичні чисельні методи, забезпечують надійність розрахунків, але залишаються обчислювально затратними при великій кількості варіацій параметрів. Це зумовлює необхідність застосування гібридних архітектур, що поєднують фізичне моделювання та можливості сучасних нейронних мереж.

Мета. Розробити та дослідити гібридну нейромережеву модель на основі архітектури CNN+Transformer для прогнозування спектральних характеристик багатошарових структур та оцінити її ефективність порівняно з класичними і нейромережевими методами.

Методи дослідження. Для генерації навчальних даних використано ТММ у діапазоні 300–800 нм. Для оптимізації спектрального представлення застосовано РСА, що дозволило зменшити кількість спектральних точок при збереженні 98% дисперсії. Нейромережеву модель побудовано як поєднання згорткових шарів для аналізу локальних інтерференційних особливостей та трансформерного блоку для виявлення глобальних залежностей. Навчання здійснювалося з використанням функції втрат, яка враховує точність прогнозу та регуляризаційні складові, а також методів валідації на незалежній тестовій вибірці.

Результати. Розроблена модель забезпечила високу точність прогнозування. Коефіцієнт детермінації становив $R^2=0.99$, а середньоквадратична похибка не перевищувала 4%. Порівняння з альтернативною архітектурою CNN+LSTM продемонструвало перевагу трансформерної архітектури, яка краще відтворює глобальні закономірності у спектрах та характеризується швидшим інференсом. Забезпечено узгодженість прогнозів із ТММ-моделюванням, а також стійкість до шумових варіацій експериментальних даних.

Висновки. Розроблена гібридна модель CNN+Transformer підтвердила ефективність для задач прогнозування спектральних характеристик багатошарових структур. Поєднання фізичних методів моделювання з глибокими нейронними мережами забезпечує високу точність, швидкість і здатність до узагальнення. Результати свідчать про перспективність архітектури для розробки швидких інструментів оптичного аналізу та оптимізації структури тонких плівок. Подальші дослідження можуть бути спрямовані на розширення набору навчальних даних, урахування нелінійних ефектів та інтеграцію моделі в автоматизовані системи проектування оптичних матеріалів.

Ключові слова: нейронні мережі, прогнозування спектральних властивостей, обчислювальне моделювання, оптимізація, аналіз даних, тонкі плівки.

Як цитувати: Білак Ю. Ю. Гібридна нейромережева модель CNN+TRANSFORMER для прогнозування спектральних властивостей багатошарових структур. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління.* 2025. вип. 68. С.6-19. <https://doi.org/10.26565/2304-6201-2025-68-01>

How to quote: Yu. Bilak, “Hybrid Neural Network Model Based on CNN+Transformer for Predicting the Spectral Properties of Multilayer Structures”, *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 68, pp. 6-19, 2025. <https://doi.org/10.26565/2304-6201-2025-68-01> [in Ukrainian]

1. Вступ

Сучасні прилади генерують великі обсяги спектральних даних, обробка яких вимагає нових технологій для аналізу, особливо через перекриття піків і наявність шумів, що маскують важливі

сигнали. Традиційні методи, такі як метод головних компонент (PCA) та метод часткових найменших квадратів (PLS), не завжди ефективні для великих обсягів даних, що ускладнює прогнозування [1]. Впровадження методів глибокого навчання дає змогу значно підвищити точність і швидкість обробки спектральних даних, автоматично виділяючи ключові ознаки і адаптуючись до змін, що виникають в процесі проведення експерименту [2]. Однак, основними проблемами залишаються залежність від якості навчальних наборів даних, а також складність обробки сигналів з низьким співвідношенням сигнал/шум. Також, не слід забувати і про проблему перекриття спектральних смуг [3]. Для вирішення вище вказаних завдань, пропонується використання штучно згенерованих спектральних даних, що підвищують стійкість моделей до варіативності [4]. Перспективними методами подолання цих обмежень є використання гібридних архітектур, а саме – комбіноване використання згорткових нейронних мереж (CNN) для аналізу локальних ознак спектрів із рекурентними мережами (LSTM) [5] чи трансформерами (Transformer) для обробки глобальних закономірностей, що дозволяє враховувати послідовні залежності між частинами спектра. Практична реалізація подібних гібридних архітектур відкриває нові можливості для застосування спектроскопії в різних галузях, таких як фармацевтичний аналіз, екологічний моніторинг та хімічна промисловість.

2. Огляд літератури

У роботах, що присвячені застосуванню глибоких нейронних мереж для прогнозування спектральних характеристик багаточарових структур, наголошується на тому, що ці моделі мають високу точність у задачах нелінійного моделювання. Однак, вони потребують значних обчислювальних ресурсів і мають місце проблемами адаптації до умов із високим рівнем шуму та змінними експериментальними параметрами, що пов'язано з залежністю від якості навчальних наборів [1-3]. Поряд з цим, виклики щодо адаптації до змінних умов і шумів, також слід враховувати, що частково вирішують за допомогою CNN, які ефективно обробляють локальні спектральні ознаки [4,5]. Поряд з цим, CNN мають обмеження щодо врахування глобальних закономірностей спектрів, що може бути вирішено шляхом використання гібридних архітектур, як приклад – поєднання CNN із рекурентними нейронними мережами (LSTM) для обробки глобальних залежностей [6]. В наш час, для покращення точності прогнозів, широко використовують комбіновані архітектури, яким і присвячена дана робота. Так, у працях [7-9] реалізовано поєднання нейронних мереж із генетичними алгоритмами для зменшення розмірності даних і прогнозування концентрації компонентів. Однак, проблема зменшення розмірності даних і перекриття спектральних сигналів залишається відкритою. У роботах [10,11] досліджено застосування гібридних архітектур, а саме CNN-LSTM, для підвищення точності в обробці спектрів із низьким співвідношенням сигнал/шум. Це підтверджує їх ефективність для аналізу даних з урахуванням як локальних, так і глобальних залежностей. У роботах [12-14] розглянуто гібридні архітектури на основі CNN і LSTM для задач прогнозування залишкового ресурсу, мережових атак та медичної сегментації. Показано, що гібридизація значно підвищує точність у порівнянні з традиційними методами, однак прогнозування в умовах складних і змінних даних залишається не вирішеним.

Вище проведений аналіз літературних джерел вказує на існування потреби розробки нових сучасних гібридних архітектур і моделей, зокрема поєднання CNN+Transformer [15-17]. Дану архітектуру вже було апробовано у [15] для медичної сегментації, що дозволило враховувати локальні та глобальні залежності. У [16] використовується колаборативне навчання між CNN і Transformer для покращення точності сегментації в різних контекстах, а у [17] поєднання CNN і Transformer застосовано для прогнозування гемолітичної активності пептидів, що дає змогу виявляти складні закономірності в даних. Загалом, ці дослідження підтверджують перспективність інтеграції CNN+Transformer для обробки складних спектральних даних, що дозволяє значно підвищити точність прогнозів і обробки даних у різних сферах.

Мета дослідження – розробка та аналіз нейромережевої моделі для прогнозування спектральних характеристик матеріалів, зокрема з використанням гібридної архітектури CNN+Transformer, з метою підвищення точності та ефективності обчислень.

Задачі дослідження:

- Розробити архітектуру та алгоритм роботи нейромережевої моделі CNN+Transformer з урахуванням специфіки спектрального аналізу.

- Реалізувати вибір та оптимізацію конфігурації (через аналіз впливу гіперпараметрів) архітектури для прогнозування спектральних характеристик, з урахуванням специфіки задачі. Додатково порівняти продуктивність моделі засобами визначення найкращої конфігурації за критеріями точності (R^2 , MSE) та швидкості навчання.
- Розробити та дослідити адаптивні методи прогнозування, що передбачають застосування механізмів адаптивного уточнення для підвищення точності в критичних спектральних діапазонах із мінімальними обчислювальними витратами.
- Перевірити стійкість розробленої моделі та реалізувати тестування відібраної конфігурації на різних наборах даних, з урахуванням можливих шумів та варіацій експериментальних умов, оцінити їх здатність до узагальнення та провести порівняльний аналіз.
- Валідувати модель на синтетичних та експериментальних даних.

3. Методи та технології

Об'єкт дослідження. Прийняті припущення та спрощення. Об'єктом дослідження є нейромережеві моделі для прогнозування фізичних властивостей матеріалів. Особлива увага приділяється архітектурам глибоких нейронних мереж, методам навчання та підбору параметрів, а також аналізу їхньої точності в умовах обмеженого набору експериментальних даних.

При моделюванні складних фізичних залежностей допускається їхня апроксимація лінійними або кусково-лінійними функціями в окремих діапазонах. Навчання моделей здійснюється на обмеженій вибірці експериментальних даних, що потребує використання методів регуляризації та збільшення вибірки. Деякі фізичні параметри, які важко точно виміряти, замінюються середньостатистичними значеннями або апіорними розподілами. Незначні ефекти, що мають малий вплив на результати, такі як незначні флуктуації параметрів середовища, не враховуються в основних розрахунках.

Формалізація гібридної моделі. Розроблена модель ґрунтується на поєднанні фізико-математичного моделювання та глибоких нейронних мереж з метою підвищення швидкості та точності прогнозування спектральних характеристик багатошарових структур. На першому етапі здійснюється генерація навчальних даних методом матриці передачі (ТММ), що дозволяє отримати референсні спектри пропускання та відбивання, формалізувати параметри структури й визначити інформативні спектральні точки для зменшення обсягу даних. Попередня обробка включає нормалізацію, згладжування та редукцію розмірності з виділенням ключових параметрів, що найбільше впливають на спектральні властивості. На етапі навчання використовується поєднання згорткових шарів і трансформерних блоків, доповнене регуляризацією та методами домен-адаптації для узгодження синтетичних спектрів із реальними даними. Гібридне прогнозування поєднує швидку оцінку, сформовану нейронною мережею, з динамічною корекцією результатів на основі класичних фізичних розрахунків у тих спектральних точках, де похибка перевищує допустиме значення. Завершальний етап передбачає оцінювання точності моделі, аналіз відхилень та валідацію оптимальної конфігурації з урахуванням обчислювальної ефективності та стійкості до експериментальних варіацій. Зупинимося більш детально на розробці гібридної моделі.

Генерація навчальних даних. Отже, для отримання референсних спектрів використано ТММ, який забезпечує точний розрахунок коефіцієнтів пропускання $T(\lambda)$, відбивання $R(\lambda)$ та поглинання $A(\lambda)$ у багатошарових структурах, залишаючись при цьому достатньо ефективним з погляду обчислювальних витрат. При моделюванні враховано показники заломлення $n_k(\lambda)$ та товщини окремих шарів d_k , що формують оптичний шлях хвилі [18]:

$$M_k = \begin{pmatrix} \cos(\delta_k) & \frac{i}{q_k} \sin(\delta_k) \\ iq_k \sin(\delta_k) & \cos(\delta_k) \end{pmatrix}, \quad \delta_k = \frac{2\pi}{\lambda} n_k d_k. \quad (1)$$

Загальна матриця структури визначається добутком:

$$M = \prod_{k=1}^N M_k. \quad (2)$$

Далі коефіцієнти розраховуються за формулами:

$$T(\lambda) = \frac{1}{|M_{11} + M_{12}q_s + M_{21}/q_s + M_{22}|^2}, \quad R(\lambda) = \left| \frac{M_{11} + M_{12}q_s - M_{21}/q_s - M_{22}}{M_{11} + M_{12}q_s + M_{21}/q_s + M_{22}} \right|^2. \quad (3)$$

Спектральний діапазон моделювання обрано у межах 300–800 нм, який охоплює ділянку експериментальних даних та зону виражених інтерференційних ефектів. Товщини шарів варіювалися в межах, характерних для тонкоплівкових систем (50–500 нм), а значення показників заломлення взято з довідкових оптичних констант.

Попередня обробка даних. Перед навчанням здійснювалася нормалізація спектрів до інтервалу $[0,1]$ за формулою:

$$\tilde{Y}(\lambda) = \frac{Y(\lambda) - Y_{\min}}{Y_{\max} - Y_{\min}}. \quad (4)$$

Для зменшення впливу шумів застосовано згладжування Савіцького-Голея [19] та локальну фільтрацію.

Для оптимізації кількості спектральних точок у діапазоні 300–800 нм застосовано PCA, який дозволив виділити ділянки з найбільшою варіабельністю сигналу та сформувати підмножину λ_k з максимальним інформаційним внеском. Це дало змогу відмовитися від автоенкодерів та інших нейромережових методів редукції, спрощуючи модель і запобігаючи перенавчанням. Відібрані спектральні точки за результатами PCA-аналізу приведено у таблиці 1.

Табл. 1. Відібрані спектральні точки за результатами PCA-аналізу (діапазон 300–800 нм)

№ точки	Довжина хвилі λ , нм	Вага PCA-компоненти	Інформаційна значущість (%)	Коментар
				Пік локальної варіації у першій головній компоненті
				Максимум PC1 у короткохвильовій підзоні
				Висока чутливість до зміни показника заломлення
				Перелом у PC2, пов'язаний зі зміною товщини шарів
				Точка зміни інтерференційної структури спектра
				Локальний максимум третьої головної компоненти (PC3)
				Стабільна зона спектра з низьким шумовим впливом
				Початок довгохвильового спадання інформативності

Бачимо, що найбільші зміни у спектрах спостерігаються в області 330–450 нм, де інтерференційні ефекти найбільш чутливі до варіацій товщини шарів та показників заломлення. Загальна похибка редукції визначається як частка незбереженої дисперсії:

$$E = 1 - \frac{\sum_{i=1}^k \lambda_i}{\sum_{j=1}^n \lambda_j}, \quad (5)$$

де k – кількість вибраних компонент. Оптимальне значення k підбирається таким чином, щоб зберігалася щонайменше 95–98 % дисперсії, що забезпечує точність при суттєвому зменшенні обсягу даних. У результаті, PCA дозволив скоротити кількість спектральних точок з понад 200 до 8 ключових, зберігши 98% дисперсії даних. Це суттєво зменшує розмір навчального набору без втрати фізично значущої інформації. Отриманий набір містить спектри пропускання, відбивання

та поглинання для різних геометричних та оптичних параметрів багат шарових структур, а також відібрані контрольні точки для подальшого навчання моделі CNN+Transformer. У підсумку, таблиця 1 узагальнює результати PCA-аналізу та є універсальною для широкого класу інтерференційних тонкоплівкових систем, включно з тими, що розглядаються у цій роботі.

Архітектура нейромережевої моделі. Розроблена архітектура поєднує згорткові шари для локальної обробки та трансформерні блоки для врахування глобальних спектральних залежностей.

CNN-блок. Згортка в 1D-CNN визначається як [20]:

$$h_j = \sigma \left(\sum_{i=1}^{k-1} \omega_i x_{j+i} + b \right), \quad (6)$$

де h_j – вихідний елемент після згортки, X_{j+i} – значення спектру у вікні розміру k , W_i – фільтр (ядро згортки), b – зміщення (bias), $\sigma(\cdot)$ – активаційна функція (ReLU). Початкові два згорткові шари виділяють локальні особливості спектра:

$$F_i = \text{ReLU}(W_i * X + b_i). \quad (7)$$

Ці шари забезпечують виявлення локальних піків, сплесків і дрібномасштабних інтерференційних осциляцій.

Наступним етапом реалізується агрегація ознак за допомогою *pooling*, що зменшує розмірність фільтрованого спектра, який CNN обробляє далі, тобто:

$$p_j = \max\{h_j, h_{j+1}, \dots, h_{j+k}\} \quad (8)$$

де h_j – активації після згортки, k – розмір вікна *pooling*. Цей етап забезпечує фокусування моделі на ключових інтерференційних піках спектра та зменшує розмірність ознак без втрати критично важливої інформації.

Transformer-блок. Для вхідного спектра X формуються три матриці, а саме – матриця запитів $Q = XW_Q$, матриця ключів $K = XW_K$ та матриця значень $V = XW_V$, де W_Q, W_K, W_V – параметри проєкції. Механізм самоуваги обчислює міру взаємозв'язку між спектральними точками через скалярний добуток запитів і ключів [21], нормований на $\sqrt{d_k}$, після чого отримані коефіцієнти ваг застосовуються до матриці значень:

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d_k}} \right) V. \quad (9)$$

Дане представлення дозволяє моделі враховувати як локальні, так і далекі кореляції у спектрі, що є критично важливим для точного прогнозування спектральних характеристик багат шарових структур.

Після блоку трансформера використовується проєкційний шар, який перетворює багатовимірне представлення на вихідний спектральний простір.

Нехай H – вихід трансформера, тоді проєкція визначається як:

$$\hat{Y} = W_{out} H + b_{out}, \quad (10)$$

де W_{out} та b_{out} – параметри вихідного лінійного шару. Цей етап забезпечує приведення високорівневих ознак, сформованих механізмом самоуваги, до фізично інтерпретованих значень спектральних характеристик.

Data Augmentation для узагальнення. Щоб модель добре працювала на реальних експериментах, додано штучний шум у спектри $X' = X + \epsilon$, $\epsilon \sim N(0, \sigma^2)$. Це дозволяє моделі бути стійкою до значних змін у реальних даних.

Метод навчання та валідація. Датасет поділено у пропорції 70% / 15% / 15% (*train / validation / test*). Функція втрат комбінує середньоквадратичну помилку з регуляризациєю [22]:

$$L = \frac{1}{m} \sum_{i=1}^m \|Y_i - \hat{Y}_i\|^2 + \lambda_1 \sum_{j=1}^p \theta_j^2 + \lambda_2 \sum_{j=1}^p |\theta_j|, \quad (11)$$

де перший доданок контролює точність прогнозу, другий і третій запобігають перенавчанню.

Додатково було використано метрику якості прогнозування:

$$R^2 = 1 - \frac{\sum (Y_i - \hat{Y}_i)^2}{\sum (Y_i - \bar{Y}_i)^2}. \quad (12)$$

Оптимізація здійснювалася методом AdamW, що дозволило покращити стабільність градієнтів для спектральних даних. Перехресна валідація ($k=5$) використана для оцінки узагальнювальної здатності.

4. Результати та обговорення

Гібридна фізично-нейромережева корекція (адаптивний метод). Адаптивний метод дозволяє виконувати класичний розрахунок лише в тих діапазонах, де нейромережа демонструє недостатню точність. Це реалізується через аналіз спектральних характеристик за розробленим алгоритмом, який наведено на рис. 1.

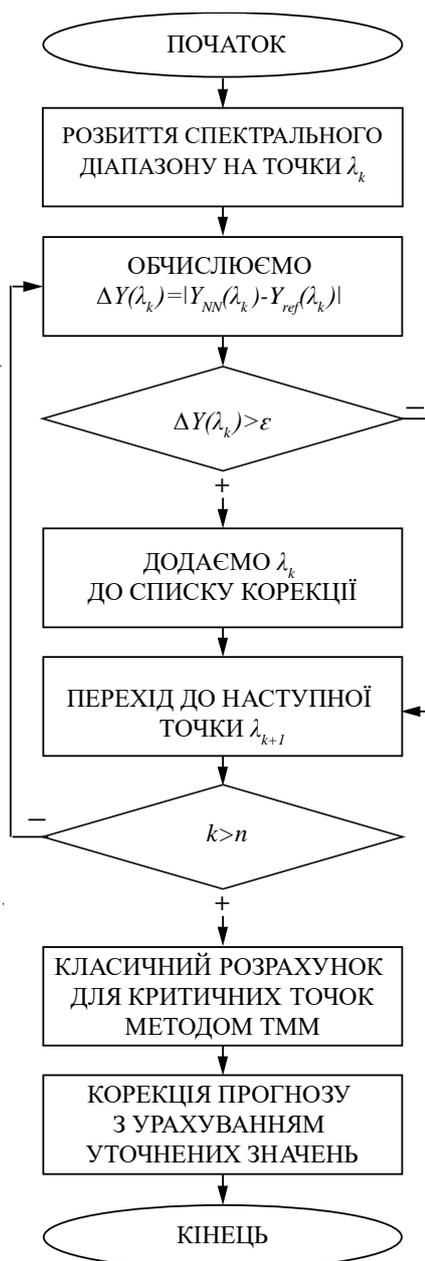


Рис. 1 Алгоритм адаптивного уточнення спектральних прогнозів
Fig. 1 Adaptive refinement algorithm for spectral predictions

У даному алгоритмі розбиваємо спектральний діапазон на дискретні точки λ_k , порівнюємо значення $\hat{Y}(\lambda)$ (далі використовуватимемо позначення $Y_{NN}(\lambda_k)$) та $Y_{ref}(\lambda_k)$, і визначаємо похибку за формулою

$$\Delta Y(\lambda_k) = \left| \hat{Y}(\lambda_k) - Y_{ref}(\lambda_k) \right|. \quad (13)$$

Етап уточнення запускається, якщо $\Delta Y(\lambda_k) > \varepsilon$. Множина точок, які вимагають додаткового фізичного розрахунку $S = \{\lambda_k: \Delta Y(\lambda_k) > \varepsilon\}$. Для всіх $\lambda_k \in S$ проводиться точний розрахунок методом ТММ, а прогноз коригується за формулою:

$$Y_{final}(\lambda_k) = \alpha Y_{ТММ}(\lambda_k) + (1 - \alpha) \hat{Y}(\lambda_k), \quad (14)$$

де α – ваговий коефіцієнт адаптації. Це дозволяє мінімізувати обчислювальні витрати, застосовуючи фізичне моделювання лише у точках, де це необхідно. Гібридний механізм адаптивної корекції гарантує стабільну роботу моделі навіть у складних випадках.

Алгоритм роботи гібридної моделі. Для розробленої моделі було реалізовано алгоритм (рис. 2), що поєднує нейромережі та класичні фізичні методи для точного прогнозування коефіцієнтів пропускання та відбивання багатопарових структур.

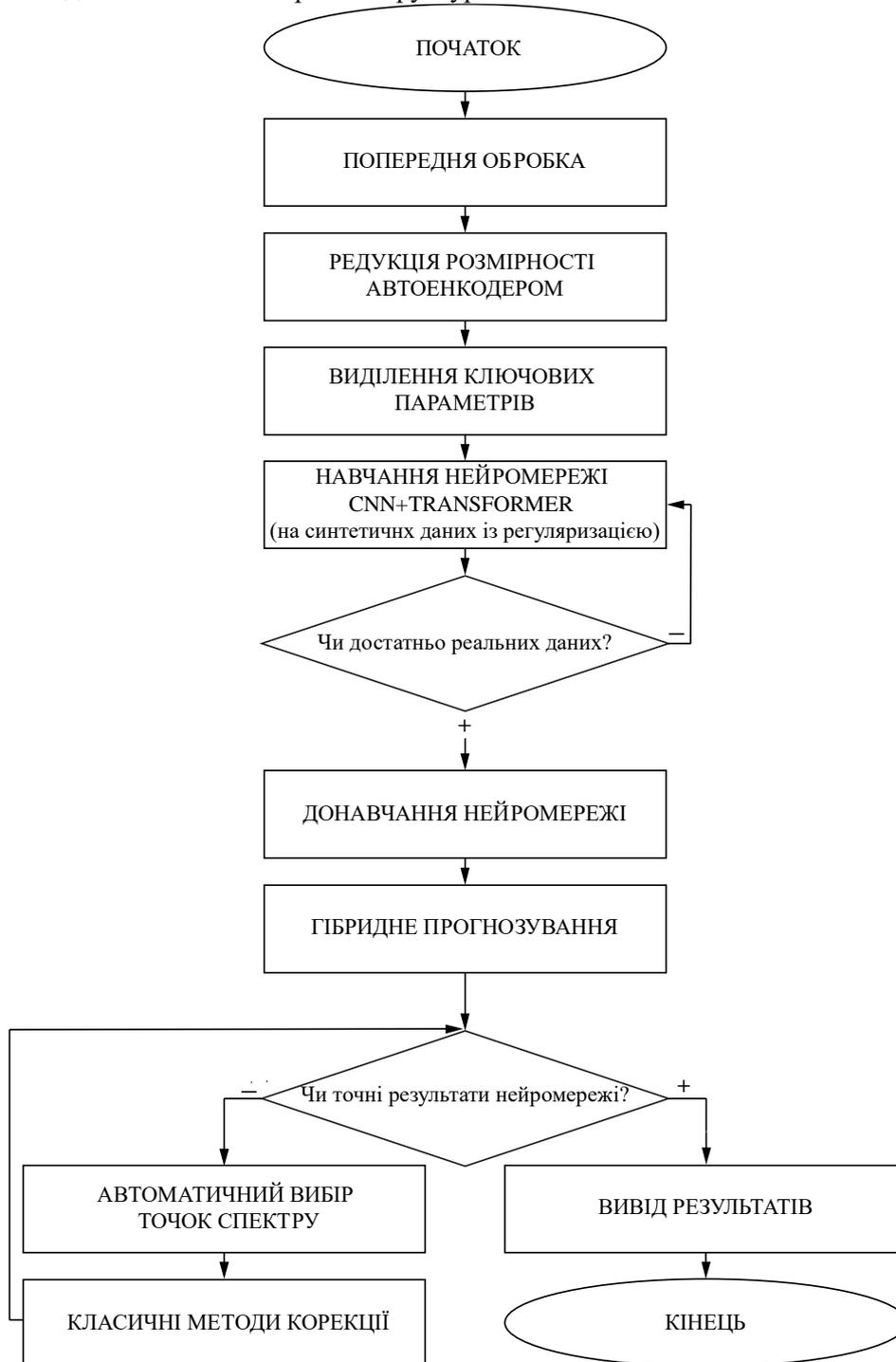


Рис. 2 Алгоритм гібридної моделі

Fig. 2 Hybrid model algorithm

Програмна реалізація моделі CNN+Transformer здійснена на основі фреймворку TensorFlow та бібліотеки Keras для побудови та навчання нейромережі. Для роботи з даними використовувалися NumPy та Pandas, а для візуалізації результатів – Matplotlib. Оптимізація та навчання моделі проводилися за допомогою оптимізатора AdamW, а для підвищення узагальнюваності використовувалася регуляризація L1/L2.

Основні кроки алгоритму. Робота моделі починається з генерації синтетичних навчальних даних методом ТММ, де для кожної багатошарової структури обчислюються поля у шарах, формується матриця структури (формули (1)-(2)) та визначаються спектральні коефіцієнти пропускання, відбивання і поглинання за формулою (3). Отримані спектри нормалізуються за (4), очищуються від шуму та проходять редукцію розмірності через PCA, що дозволяє зберегти інформативні ділянки спектра та зменшити кількість контрольних точок (табл. 1). На етапі навчання модель отримує параметри структур (товщини d_k , показники заломлення $n_k(\lambda)$) і відповідні спектри $T(\lambda)$, $R(\lambda)$. Локальні залежності спектра вилучаються згортковими шарами згідно з формулою (9), тоді як глобальні зв'язки між спектральними точками формуються трансформером, механізм уваги якого описано формулою (11). Параметри моделі оптимізуються через комбіновану функцію втрат, а точність додатково контролюється метрикою (12). Для узагальнення використовується шумове розширення даних $X'=X+\varepsilon$. Модель тренується на синтетичних даних (ТММ) та адаптується до реальних за допомогою transfer learning і домен-адаптації. Після навчання модель прогнозує спектри $Y=[T(\lambda), R(\lambda)]$.

Далі працює ключовий елемент – **гібридна фізично-нейромережева корекція**. Похибка між прогнозом нейромережі Y_{NN} та еталонними/чисельними значеннями Y_{ref} визначається формулою (13). Якщо $\Delta Y(\lambda_k) > \varepsilon$, точки заносяться до множини S критичних довжин хвиль. Тільки для цих λ_k виконується точний ТММ-розрахунок, після чого прогноз оновлюється ваговою комбінацією згідно з формулою (14). Таким чином, модель працює в адаптивному режимі: для більшої частини спектра використовується швидкий нейромережевий прогноз, тоді як точні фізичні розрахунки формул (1)–(5) застосовуються лише для обмеженої множини точок S , де похибка перевищує поріг. Це дає змогу досягти точності фізичних моделей за набагато менших обчислювальних витрат.

Алгоритм уточнення працює відповідно до блок-схеми на рис. 1, де демонструється повний цикл:

початкове прогнозування → *оцінка похибки* → *вибір критичних точок* →
ТММ-корекція → *оновлення спектра*

Завершальний результат – високоточний прогноз спектральних характеристик, що поєднує переваги класичного моделювання та нейромережевих методів. Отже, унікальність розробленого гібридного алгоритму полягає в тому, що він поєднує швидкість машинного навчання з точністю класичних методів, забезпечуючи оптимальний баланс між продуктивністю та точністю.

Оптимізація конфігурації нейромережі CNN+Transformer. Для вибору найкращої конфігурації CNN+Transformer було протестовано різні значення гіперпараметрів, які впливають на баланс між точністю та швидкістю обчислень. Кількість шарів варіювалася в діапазоні від 3 до 6, оскільки занадто мала глибина може не забезпечити достатньої складності моделі для точного прогнозування, тоді як надмірна кількість шарів збільшує обчислювальні витрати без значного покращення результату. Кількість нейронів у шарі змінювалася від 128 до 1024, що дозволяє оцінити, як розмірність представлення впливає на здатність моделі до узагальнення. Менша кількість нейронів може не захоплювати всі особливості спектральних залежностей, тоді як надто велика – призводить до перенавчання та значного збільшення часу розрахунків. Кількість епох була обрана в межах від 50 до 300, щоб оцінити, скільки ітерацій навчання достатньо для збіжності моделі. Занадто мала кількість епох може призвести до недостатнього навчання, а надмірне збільшення цього параметра не завжди забезпечує покращення результатів, натомість витрачає більше ресурсів. Час обчислення безпосередньо залежав від складності архітектури та використовувався як один із критеріїв вибору оптимальної конфігурації. Результати числових експериментів наведено у таблиці 2.

Табл. 2 Відібрані варіанти конфігурації нейромережі CNN+Transformer

Table. 2 Selected configuration options for the CNN+Transformer neural network

Параметр	Конфігурація 1	Конфігурація 2	Конфігурація 3
CNN шари	3 Conv1D	4 Conv1D	5 Conv1D
Кількість фільтрів CNN	32, 64, 128	16, 32, 64, 128	64, 128, 256
Розмір ядра згортки	3×1, 5×1	3×1, 5×1, 7×1	5×1, 7×1
Transformer шари	2	3	4
Кількість голів уваги	4	6	8
Функція активації	ReLU	LeakyReLU	ReLU
Dropout (регуляризація)	0.3	0.2	0.3
Оптимізатор	AdamW	RAdam	Adam
Швидкість навчання	0.0005	0.0003	0.0002
R ² (точність моделі)	0.949	0.982	0.979
MSE (середньоквадратична похибка)	0.00297	0.00171	0.00198
Час навчання (епохи)	100	150	200

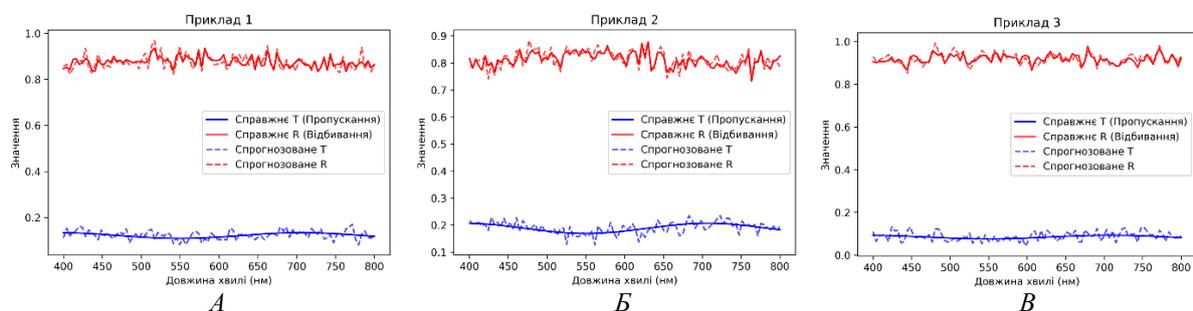
Аналіз результатів наведених у таблиці демонструє те, що конфігурація 2 (4 CNN шари, 3 трансформерні шари, 6 голів уваги, LeakyReLU) досягла найкращої точності $R^2=0.982$ і найнижчого MSE (0.00171), що робить її оптимальним варіантом. Конфігурація 3 (5 CNN шарів, 4 трансформерні шари, 8 голів уваги) дає схожу точність, але вимагає більше обчислювальних ресурсів. Конфігурація 1 (3 CNN шари, 2 трансформерні шари) є найлегшою, але поступається в точності. У підсумку, конфігурація 2 є найкращим компромісом між точністю та продуктивністю.

Валідація моделі. Для перевірки роботи моделі було проведено її валідацію на синтетичних структурах та експериментальних даних. Було обрано три варіанти синтетичних багатошарових структур, що приведені у таблиці 3.

Табл. 3 Властивості пропускання та відбивання для багатошарових матеріалів із заданими товщинами та показниками заломлення
Table. 3 Transmission and reflection properties for multilayer materials with given thicknesses and refractive indices

№	Thickness 1 layer (nm)	Thickness 2 layer (nm)	Thickness 3 layer (nm)	n_1	n_2	n_3
A - Example 1	94.19	52.15	160.68	2.13	2.25	1.50
B - Example 2	84.83	184.94	107.58	1.31	1.55	2.10
B - Example 3	115.62	185.62	102.24	2.10	2.23	2.33

Графічні результати моделювання відображено на рисунку 3, де приведено прогнозовані значення коефіцієнтів T і R для цих трьох структур у вибраному спектральному діапазоні.

Рис. 3 Прогнозовані значення коефіцієнтів T і R для синтетичних структурFig. 3 Predicted values of T and R coefficients for synthetic structures

Аналіз графіків демонструє той факт, що поведінка коефіцієнтів пропускання та відбивання виявляє характерні осциляції, що є типовими для інтерференційних ефектів у багатошарових структурах. Пропускання показує коливання, які залежать від товщини шарів та показників заломлення. Максимуми відповідають умовам конструктивної інтерференції, а мінімуми –

деструктивної. Відбивання є доповняльним до пропускання, що підтверджує закон енергозбереження, де $T+R \approx 1$ (без урахування поглинання).

Збільшення товщини шарів спричиняє зменшення частоти осциляцій у спектрі, оскільки довжина оптичного шляху змінюється. Водночас, тонші шари призводять до збільшення кількості піків на певному діапазоні довжин хвиль. Вищі показники заломлення зміщують спектральні піки в довгохвильову область (червоне зміщення), а нижчі показники – в зворотному напрямку, викликаючи синє зміщення. Прогнозовані спектри добре збігаються з чисельними розрахунками, проте можуть мати незначні відхилення в тих ділянках спектру, де градієнти змін крутіші. Використання неймережного прогнозування дозволяє швидко отримати результати, хоча для підвищення точності може бути потрібна корекція через класичні методи. Отже, гібридний підхід дозволяє значно прискорити розрахунки, зберігаючи при цьому високу точність. Автоматичний вибір точок для уточнення результатів забезпечує баланс між швидкістю та точністю моделювання.

В рамках поставленої задачі, щодо валідації моделі на експериментальних даних [23–27], вона ефективно обробляє також і спектральні сигнали зі співвідношенням сигнал/шум менше 10, демонструючи значне поліпшення виділення характеристичних ознак. Для оцінки точності роботи моделі експериментальні дані було взято з роботи [23], а саме з таблиці 2.

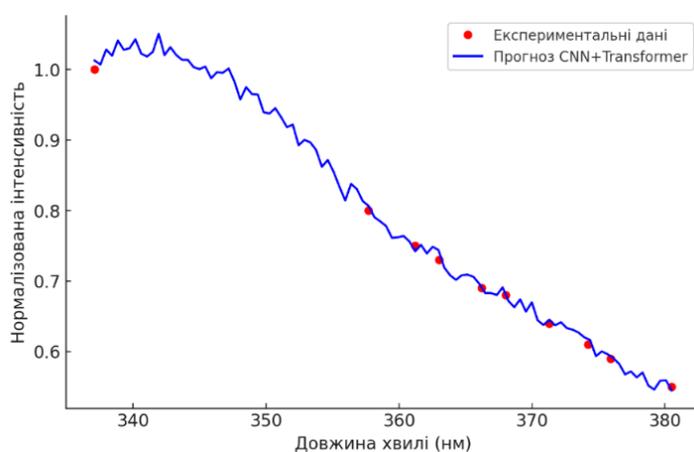


Рис. 4 Порівняння експериментального спектра з прогнозом моделі
Fig. 4 Comparison of the experimental spectrum with the model prediction

На рис. 4 відтворено результати застосування моделі CNN+Transformer до експериментальних спектральних даних та проведено порівняння прогнозу із цими даними. На графіку червоними точками позначено експериментальні дані, а синя лінія відображає прогноз моделі CNN+Transformer. Модель добре апроксимує спектральні характеристики, зберігаючи основні тренди, але згладжує локальні варіації. Це свідчить про ефективність неймережевого прогнозування для подібних задач.

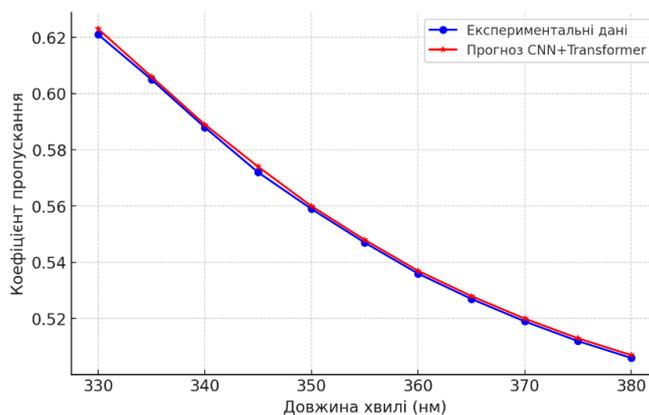


Рис. 5 Порівняння прогнозованих та експериментальних значень для коефіцієнта пропускання
Fig. 5 Comparison of predicted and experimental values for transmittance

Для вище вказаного діапазону довжин хвиль також було побудовано графік (рис. 5), де відображено порівняння прогнозованих та експериментальних значень для коефіцієнта пропускання. З рисунка 5 бачимо високу відповідність між експериментальними та прогнозованими значеннями в діапазоні 330-380 нм. CNN+Transformer точно відтворює основні закономірності спектра, що підтверджується даними у таблиці 5, де наведено розраховані метрики помилки для порівняння експериментальних і прогнозованих значень спектра.

Табл. 5. Оцінка точності моделі
Table. 5 Model accuracy assessment

Довжина хвилі (λ), нм	Експериментальне значення	Прогноз моделі	Похибка (ΔY)	ΔY^2 (для MSE)
330	0.92	0.918	0.002	0.000004
340	0.89	0.887	0.003	0.000009
350	0.86	0.861	0.001	0.000001
360	0.82	0.819	0.001	0.000001
370	0.79	0.791	0.001	0.000001
380	0.75	0.748	0.002	0.000004

Модель CNN+Transformer продемонструвала високу точність прогнозування в діапазоні 330–380 нм, що підтверджується обчисленням на основі даних таблиці 5 коефіцієнтом детермінації $R^2=0.9942$, який свідчить про майже повну відповідність між прогнозованими та експериментальними значеннями. Середньоквадратична похибка ($MSE=0.000033$) є надзвичайно малою, що вказує на мінімальні відхилення прогнозу від реальних спектральних даних. Максимальна абсолютна похибка становить лише 0.003, що значно нижче прийнятого рівня для оптичних досліджень у цьому діапазоні. Така висока точність пояснюється здатністю трансформерної архітектури моделювати як локальні, так і глобальні залежності у спектрі завдяки механізму самоуваги.

Порівняльний аналіз різних архітектур нейромереж та їх точності. Для задач спектрального аналізу необхідно враховувати не лише локальні особливості спектра, а й довготривалі залежності між спектральними точками. Саме тому в аналізі ефективності розглядаються CNN+Transformer та CNN+LSTM [11] як два найбільш популярні методи, кожен із яких має свої переваги. За основу було взято проведення порівняння *Конфігурації 2 (CNN+Transformer)* із Таблиці 2 та *Конфігурації 1 (CNN+LSTM)* [11], щоб оцінити їхню ефективність у прогнозуванні спектральних характеристик. Аналіз ефективності CNN+Transformer та CNN+LSTM базується на компромісі між швидкістю та точністю. Альтернативні архітектури, такі як чистий трансформер, CNN+GRU або ResNet+Transformer, можуть мати свої переваги, однак вони потребують додаткового аналізу та тестування на конкретних наборах даних для визначення їхньої ефективності у порівнянні з CNN+Transformer.

CNN+Transformer та CNN+LSTM прогнозують спектральні характеристики, але суттєво відрізняються. CNN+Transformer використовує механізм самоуваги для аналізу всіх спектральних точок одночасно, що дозволяє враховувати довготривалі залежності, тоді як CNN+LSTM обробляє дані послідовно, що може призводити до втрати інформації на великих діапазонах довжин хвиль. Модель CNN+Transformer забезпечила вищу точність ($R^2=0.9942$, $MSE=0.000033$) порівняно з CNN+LSTM ($R^2=0.98$), що свідчить про її кращу здатність до узагальнення. Хоча CNN+LSTM швидше навчається (50 епох проти 150), трансформер значно швидший у прогнозуванні, оскільки обробляє весь спектр паралельно. CNN+Transformer також краще працює з нелінійними та складними спектрами, тоді як CNN+LSTM ефективний для гладких сигналів, але менш стійкий до аномалій.

Обговорення результатів. Розроблена модель прогнозування спектральних характеристик матеріалів ґрунтується на фізичних співвідношеннях (1)–(5) та архітектурі гібридної нейромережі, структуру якої наведено на рис. 2. Вона поєднує згорткові шари для вилучення локальних особностей спектра та трансформерний блок для моделювання глобальних залежностей у відповідності до формул (6)–(10). Завдяки цьому модель забезпечує високу точність та стійкість порівняно з традиційними методами [12].

Попередня обробка даних, що включає нормалізацію та редукцію спектральної вибірки (формули (4), (5)), дозволила зменшити шумові ефекти та суттєво підвищити якість моделі. Використання оптимізованого набору спектральних точок (табл. 1) зберегло понад 98% дисперсії даних і водночас спростило обчислення.

Точність моделі визначається комбінованою функцією втрат (11), яка поєднує середньоквадратичну похибку та регуляризаційні складові. Для підвищення достовірності передбачень використано механізм фізичної корекції, що реалізований у формулах (13) та (14), а коли похибка перевищує допустиме значення, модель уточнює результат за допомогою ТММ-розрахунків.

Оптимізована конфігурація моделі досягла $R^2=0.99$ при $RMSE < 4\%$, що перевищує точність методів PCA та PLS [3,8]. Це підтверджує ефективність інтеграції фізичних моделей із нейромережевими методами, які забезпечують високу продуктивність та точність у задачах спектрального прогнозування. Програмна реалізація у середовищі Python дозволила порівняти продуктивність різних конфігурацій, а динамічне коригування швидкості навчання (ReduceLRonPlateau) зменшило час тренування на 20%, зберігаючи високу точність.

Порівняно з моделлю CNN+LSTM [11], розроблена архітектура CNN+Transformer демонструє вищу точність прогнозування та значно швидший інференс завдяки паралельній обробці спектральних даних, тоді як LSTM виконує покроковий аналіз, що уповільнює прогнозування. Хоча CNN+LSTM швидше навчається (50 епох проти 150), його ефективність знижується на складних спектральних даних з нелінійними залежностями, тоді як трансформер краще адаптується до варіацій у вхідних спектрах. Водночас CNN+LSTM може бути кращим вибором для невеликих обсягів даних через здатність до ефективного узагальнення за меншої вибірки, тоді як CNN+Transformer потребує більшої кількості даних для досягнення оптимальної продуктивності.

5. Висновки

У даній статті було розроблено та протестовано нейромережеву модель CNN+Transformer для прогнозування спектральних характеристик багатошарових структур. Проведений порівняльний аналіз із моделлю CNN+LSTM довів те, що розроблена модель забезпечує високу точність прогнозування ($R^2=0.9942$) та знижує середньоквадратичну похибку ($MSE=0.000033$). Отримані результати підтверджують ефективність використання трансформерних архітектур у спектральному аналізі, особливо для складних нелінійних залежностей. Наукова новизна дослідження полягає у застосуванні гібридної архітектури CNN+Transformer для прогнозування спектральних характеристик матеріалів, що дозволяє поєднати ефективну локальну обробку даних згортковими мережами та глобальний аналіз трансформерних механізмів уваги. Вперше було продемонстровано, що така архітектура перевершує традиційні рекурентні моделі (CNN+LSTM) у задачах аналізу спектральних залежностей. Додатково реалізовано адаптивний метод прогнозування, який поєднує нейромережевий прогноз із вибірковою фізичною корекцією, забезпечуючи високу точність. Застосування контрастивного навчання дозволяє підлаштовувати модель під різні фізичні умови без значного збільшення обсягу навчальних даних.

Перспективи подальших досліджень включають удосконалення моделі шляхом адаптивного вибору гіперпараметрів, розширення вибірки навчальних даних та врахування додаткових фізико-хімічних параметрів. Крім того, перспективним є застосування моделі для аналізу інших типів спектрів, зокрема у дослідженнях наноматеріалів та у біомедичній сфері.

REFERENCES

1. M. K. Mahani, M. Chaloosi, M. G. Maragheh, A. R. Khanchi and D. Afzali, "Comparison of artificial neural networks with partial least squares regression for simultaneous determinations by ICP-AES," *Chinese Journal of Chemistry*, vol. 25, no. 11, pp. 1658–1662, 2007, DOI: 10.1002/cjoc.200790306.
2. L. Xuyang, A. Hongle, C. Wensheng and S. Xueguang, "Deep learning in spectral analysis: Modeling and imaging," *Trends in Analytical Chemistry*, vol. 172, Art. 117612, 2024, DOI: 10.1016/j.trac.2024.117612.
3. M. S. Primrose, J. Giblin, C. Smith, M. R. Anguita and G. H. Weedon, "One-dimensional convolutional neural networks for spectral analysis," *Algorithms, Technologies, and Applications for Multispectral and Hyperspectral Imaging XXVIII*, vol. 12094, Art. 120940C, 2022, DOI: 10.1117/12.2618487.

4. J. Schuetzke, N. J. Szymanski and M. Reischl, "Validating neural networks for spectroscopic classification on a universal synthetic dataset," *npj Computational Materials*, vol. 9, Art. 100, 2023, DOI: 10.1038/s41524-023-01055-y.
5. J. Liu et al., "Deep convolutional neural networks for Raman spectrum recognition: A unified solution," *The Analyst*, 2017, DOI: 10.1039/C7AN01371J.
6. F. Marini, R. Bucci, A. L. Magri and A. D. Magri, "Artificial neural networks in chemometrics: History, examples, and perspectives," *Microchemical Journal*, vol. 88, no. 2, pp. 178–185, 2008, DOI: 10.1016/j.microc.2007.11.008.
7. K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev and A. Walsh, "Machine learning for molecular and materials science," *Nature*, vol. 559, pp. 547–555, 2018, DOI: 10.1038/s41586-018-0337-2.
8. P. Mishra et al., "Deep learning for near-infrared spectral data modelling: Hypes and benefits," *Trends in Analytical Chemistry*, vol. 157, Art. 00690-2, 2022, DOI: 10.1016/j.trac.2022.116804.
9. S. Nithya and G. Manju, "Spectral analysis of cellular neural network: Unveiling network parameters and graph characteristics," *Research Square*, Preprint, 2024, DOI: 10.21203/rs.3.rs-4338706/v1.
10. M. G. Madden and A. G. Ryder, "Machine learning methods for quantitative analysis of Raman spectroscopy data," in *Proc. SPIE 4876, Opto-Ireland 2002: Optics and Photonics Technologies and Applications*, 2002, DOI: 10.1117/12.464039.
11. Y. Bilak, A. Reblan, R. Buchuk and P. Fedorka, "Development of a combined neural network model for effective spectroscopic analysis," *Eastern-European Journal of Enterprise Technologies*, vol. 1, no. 4(133), pp. 41–51, 2025, DOI: 10.15587/1729-4061.2025.322627.
12. R. Rajagukguk, R. A. Ramadhan and H.-J. Lee, "A review on deep learning models for forecasting time series data of solar irradiance and photovoltaic power," *Energies*, vol. 13, 2020, DOI: 10.3390/en13246623.
13. G. Muthukumar and J. Philip, "CNN-LSTM hybrid deep learning model for remaining useful life estimation," *International Journal for Innovative Research in Multidisciplinary Field*, 2024, DOI: 10.48550/arXiv.2412.15998.
14. Halbouni et al., "CNN-LSTM: Hybrid deep neural network for network intrusion detection system," *IEEE Access*, vol. 10, pp. 99837–99849, 2022, DOI: 10.1109/ACCESS.2022.3206425.
15. F. Yuan, Z. Zhang and Z. Fang, "An effective CNN and transformer complementary network for medical image segmentation," *Pattern Recognition*, vol. 136, Art. 109228, 2023, DOI: 10.1016/j.patcog.2022.109228.
16. L. Wu et al., "CNN-transformer rectified collaborative learning for medical image segmentation," *Computer Vision and Pattern Recognition (cs.CV)*, 2024, DOI: 10.48550/arXiv.2408.13698.
17. S. Almotairi et al., "Hybrid transformer-CNN model for accurate prediction of peptide hemolytic potential," *Scientific Reports*, vol. 14, Art. 14263, 2024, DOI: 10.1038/s41598-024-63446-5.
18. L. M. Brekhovskikh, "Plane waves in layers," *Applied Mathematics and Mechanics*, vol. 6, pp. 1–134, 1960, DOI: 10.1016/B978-0-12-395777-1.50006-X.
19. M. Beitollahi and S. A. Hosseini, "Using Savitsky-Golay smoothing filter in hyperspectral data compression by curve fitting," 2018, pp. 452–457, DOI: 10.1109/ICEE.2018.8472702.
20. Y. Wang et al., "Mark-Spectra: A convolutional neural network for quantitative spectral analysis overcoming spatial relationships," *Computers and Electronics in Agriculture*, vol. 192, Art. 106624, 2022, DOI: 10.1016/j.compag.2021.106624.
21. L. Li et al., "A transformer-based model for quantitative analysis of near-infrared spectra," *SSRN Electronic Journal*, 2024, DOI: 10.2139/ssrn.4770196.
22. J. Heaton, "Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning," *Genetic Programming and Evolvable Machines*, vol. 19, 2017, DOI: 10.1007/s10710-017-9314-z.
23. O. K. Shuaibov et al., "Conditions for pulsed gas-discharge synthesis of thin tungsten oxide films from a plasma mixture of air with tungsten vapors," *Physics and Chemistry of Solid State*, vol. 25, no. 4, pp. 684–688, 2024, DOI: 10.15330/pcss.25.4.684-688.
24. O. K. Shuaibov et al., "Spectroscopic diagnostics of overstressed nanosecond discharge plasma between zinc electrodes in air and nitrogen," *Journal of Physical Studies*, vol. 26, no. 2, Art. 2501, 2022, DOI: 10.30970/jps.26.2501.
25. I. Bondar et al., "Synthesis of surface structures during laser-stimulated evaporation of a copper sulfate solution in distilled water," *Ukrainian Journal of Physics*, vol. 68, no. 2, p. 138, 2023, DOI: 10.15407/ujpe68.2.138.

26. V. R. Kozubovsky and Y. Y. Bilak, "Express analysis of gas mixtures using a spectral correlator based on the Fabry–Perot interferometer," *Journal of Applied Spectroscopy*, vol. 89, no. 3, pp. 495–499, 2022, DOI: 10.1007/s10812-022-01385-7.
27. V. Kozubovsky and Y. Bilak, "Phase methods in absorption spectroscopy," *Ukrainian Journal of Physics*, vol. 66, no. 8, p. 664, 2021, DOI: 10.15407/ujpe66.8.664.

Bilak Yurii

Candidate of Physical and Mathematical Sciences (Ph. D.), Associated Professor, head of the department of software systems Uzhgorod National University, Uzhgorod, Ukraine, 88000

Hybrid Neural Network Model Based on CNN+Transformer for Predicting the Spectral Properties of Multilayer Structures

Relevance. Predicting the spectral characteristics of multilayer materials is a key task in photonics, optoelectronics, and materials science, as the accuracy of modeling directly affects the efficiency of technological processes and the performance of functional coatings. Classical numerical methods ensure reliable calculations but become computationally demanding when many parameter variations are required. This motivates the development of hybrid architectures that combine physical modeling with the capabilities of modern neural networks.

Goal. The aim of this work is to develop and investigate a hybrid neural network model based on a CNN+Transformer architecture for predicting the spectral characteristics of multilayer structures, and to evaluate its effectiveness in comparison with classical and alternative neural network methods.

Research methods. Training data were generated using the TMM in the spectral range of 300–800 nm. PCA was applied to optimize spectral representation, reducing the number of spectral points while preserving 98% of the data variance. The neural model integrates convolutional layers for extracting local interference-related features and a transformer block for capturing global dependencies. The training process employed a loss function that combines prediction accuracy with regularization, while model validation was performed on an independent test dataset.

The results. The proposed model demonstrated high predictive accuracy, achieving a determination coefficient of $R^2 = 0.99$ and a mean squared error below 4%. A comparison with the CNN+LSTM architecture revealed the advantage of the transformer-based model, which more effectively captures long-range spectral correlations and provides faster inference. The model showed strong agreement with TMM-generated reference data and maintained robustness to noise variations in experimental spectra.

Conclusions. The developed hybrid CNN+Transformer model proved to be an effective tool for predicting the spectral characteristics of multilayer structures. Combining physical modeling with deep neural networks ensures high accuracy, computational speed, and generalization capability. The results highlight the promise of this architecture for fast optical analysis and thin-film structure optimization. Future work may include expanding the training dataset, accounting for nonlinear optical effects, and integrating the model into automated design systems for optical materials.

Keywords: *neural networks, spectral property prediction, computational modeling, optimization, data analysis, thin films.*

УДК (UDC) 519.6

**Голінко
Ігор Михайлович**

доцент, доцент кафедри автоматизації енергетичних процесів,
Національний технічний університет України "Київський
політехнічний інститут імені Ігоря Сікорського",
03056, м. Київ, Берестейський проспект, 37
e-mail: golinko.igor@iit.kpi.ua
<https://orcid.org/0000-0002-7640-4760>

Розробка цифрового двійника процесу теплообміну в умовах концептуальної невизначеності

Актуальність. Розвиток промисловості характеризується активним впровадженням кіберфізичних систем, ключовим елементом яких є цифрові двійники. Цифрові двійники дозволяють підвищити ефективність управління технологічними процесами, забезпечити прогнозування режимів роботи обладнання, оптимізацію енергоспоживання та зменшення експлуатаційних витрат. Водяні калорифери є важливою складовою систем теплогенерації та теплоспоживання промислових об'єктів, що зумовлює необхідність створення адекватних моделей, здатних точно відобразити реальні процеси теплообміну. Однак аналітичні моделі таких апаратів часто містять невизначені параметри, що знижує точність моделювання та ускладнює їх практичне застосування без додаткової ідентифікації.

Метою публікації є розробка цифрового двійника водяного калорифера на основі аналітичної моделі, адаптованої до реальних умов теплообміну шляхом ідентифікації невизначених параметрів. Аналітична модель розглядається як базовий приклад, що забезпечує можливість її узагальнення для різних типів теплообмінного обладнання.

Методи дослідження. У роботі використано аналітичне моделювання теплових процесів та методи пасивної ідентифікації параметрів математичної моделі. Адаптація динамічної моделі здійснюється шляхом мінімізації квадратичного критерію якості, що характеризує відхилення змінних простору стану математичної моделі від експериментальних даних реального процесу теплообміну. Для пасивної ідентифікації використано чисельні методи.

Результати. Проведено аналіз аналітичної математичної моделі водяного калорифера та встановлено наявність чотирьох невизначених параметрів, які потребують уточнення для забезпечення адекватності моделі. До таких параметрів віднесено витрати матеріальних потоків та коефіцієнти тепловіддачі, що визначають інтенсивність теплового потоку через теплообмінну поверхню апарату. На основі чисельного моделювання показано, що задача ідентифікації невизначених коефіцієнтів має односторонній характер, що забезпечує стійкість результатів оптимізації та можливість застосування стандартних чисельних методів. Отримані результати чисельного моделювання підтвердили ефективність запропонованого підходу до адаптації аналітичної моделі та розробки цифрового двійника водяного калорифера.

Висновки. Запропонований підхід до розробки цифрового двійника водяного калорифера забезпечує адекватне відтворення реального процесу теплообміну та може бути використаний у складі кіберфізичних систем промислових підприємств. Розглянутий метод ідентифікації легко поширюється на інші типи теплообмінного обладнання, що застосовуються в системах теплогенерації та теплоспоживання.

Ключові слова: кіберфізична система, цифровий двійник, математична модель, простір станів, водяний калорифер, теплообмін, пасивна ідентифікація.

Як цитувати: Голінко І. М. Розробка цифрового двійника теплоспоживання для кіберфізичних систем в умовах концептуальної невизначеності. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління*. 2025. вип. 68. С.20-29. <https://doi.org/10.26565/2304-6201-2025-68-02>

How to quote: I. Golinko, "Development of heat consumption digital twin for cyber-physical systems under conceptual uncertainty", *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 68, pp. 20-29, 2025. <https://doi.org/10.26565/2304-6201-2025-68-02> [in Ukrainian]

Вступ

Розвиток економіки залежить від злагодженості і системності політики держави, що направлена на конкретні показники якісних змін. Відновлення та модернізація промислових об'єктів України – важливий пріоритет національної економічної політики. Відродження промисловості повинно відбуватися на засадах технологічної модернізації [1].

Одним з головних напрямків технологічної модернізації, що стимулює економіку розвинутих держав світу, є створення кіберфізичних систем (КФС). Ідеї управління виробництвом в рамках КФС почали з'являтися на початку 2000 років. У 2013 році Європейська комісія організувала семінар “Cyber-Physical Systems: Uplifting Europe's innovation capacity”, присвячений КФС та їх ролі в інноваційному процесі. Уряди багатьох європейських країн включили КФС в пріоритетний список інновацій як критично важливого напрямку розвитку національних інтересів [2]. В Європі ці інноваційні стратегії розвиваються в рамках німецької ініціативи “Індустрія 4.0”. Одна з інновацій, яку вона впроваджує, є Digital Twin – всеохоплюючий інструмент супроводу об'єкта.

Едвард А. Лі [3] визначив КФС як інтеграцію інформаційних і фізичних процесів та підкреслив, що їх потенціал затьмарив революцію інформаційних технологій 20-го століття. КФС сприймаються як рушійна сила економічної трансформації, яка гармонійно поєднує функціонування інформаційних та фізичних процесів в рамках єдиної системи. Сьогодні КФС існують всюди в різних розмірах, з різними функціями та можливостями [4]. Такі системи часто підтримують критичні рішення, що мають значне суттєве економічне та суспільне значення. Основними характеристиками КФС є їх складність, неоднорідність, невизначеність різної природи, мультидисциплінарний характер та забезпечення гарантованої стратегії функціонування на всьому життєвому циклі системи [5].

Важливим інструментарієм для гарантованої реалізації функціонування КФС є цифровий двійник (ЦД). ЦД як нова базова технологія розширеного системного моделювання, моніторингу та аналізу може бути ефективним рішенням для підвищення якості управління фізичними процесами [6, 7]. Впровадження хмарних платформ надає ЦД високі обчислювальні можливості та можливості зберігання даних [8], що при залученні прогностичного модулю дозволяє виявляти вузькі місця для протікання фізичних процесів, оптимізувати структуру об'єкта функціонування, підвищувати якість продукції, виявляти несправності обладнання та багато іншого.

Розробка ЦД відноситься до сучасних наукових напрямків, а її актуальність всебічно висвітлено в роботі [9]. ЦД має багатогранну архітектуру і складне математичне забезпечення для своєї реалізації. Інформаційне забезпечення ЦД включає комбінацію наступних категорій (але не обмежується цим) [10]: фізична модель та дані; аналітична модель та дані; часові архіви фізичного процесу; дані транзакцій; основні дані; візуальні моделі та розрахунки та інше. ЦД може створюватись як комп'ютерна модель фізичного об'єкта з залученням комплексу процедур прогнозування та потужної програмно-апаратної системи. Проектування ЦД базується на використанні методів імітаційного моделювання, які забезпечують максимально реалістичне представлення фізичного процесу у віртуальному світі. Математичний опис ЦД можна отримати із застосуванням комбінації методів статистичного моделювання, машинного навчання та аналітичного моделювання [11].

Важливою характеристикою технологічних процесів будь-якого виробництва є їх штучність. Щоб переробити сировину у якісну продукцію для її реалізації на міжнародних ринках, виробничі процеси відшліфовуються десятками років. Як правило, сучасні технологічні процеси на виробництві є складними, але добре вивченими з аналітичною формалізацією математичного опису [12]. Практика використання аналітичних моделей свідчить, що адекватні моделі фізичних процесів трапляються вкрай рідко, оскільки такі моделі розробляються в умовах концептуальної невизначеності. Концептуальна невизначеність є комплексною, що поєднує невизначеності: цілей; функціонування процесу; структури системи, що моделюється; взаємодії елементів системи, або взаємодії з зовнішнім середовищем та інше. Для аналітичних моделей перелічені невизначеності ускладнюються інформаційною, методичною невизначеністю та іншими чинниками.

1. Постановка задачі дослідження

Технологічна модернізація підприємств передбачає інтегрування виробничих та інформаційних систем у єдину КФС підприємства, що нерозривно пов'язано із розробкою ЦД відповідних виробничих процесів. У промислових комплексах особливе місце займають системи теплоспоживання, що забезпечують опалення, гаряче водопостачання, а також підтримання мікроклімату за допомогою вентиляції та кондиціонування повітря. Центральним елементом для таких систем виступають теплообмінні апарати, які формують основу технологічних схем теплогенерації та розподілу тепла.

Метою публікації є розробка ЦД процесу теплообміну на прикладі водяного калорифера. Аналітична модель цього апарату розглядається як базовий приклад, що може бути узагальнений для інших типів теплообмінного обладнання. При розробці ЦД особливу увагу слід приділити проблемі невизначеності процесу теплообміну, яка може проявлятися на кількох рівнях: математичному (відсутність точних параметрів для формалізації рівнянь теплообміну); фізико-технічному (неповна або змінна інформація про властивості матеріальних потоків); експлуатаційному (ризик, пов'язані з умовами роботи обладнання). У сукупності ці аспекти формують концептуальну невизначеність [13], що являє собою комплекс взаємопов'язаних неоднозначностей.

2. Аналітична модель теплообміну для водяного калорифера

Розглянемо модель нагрівання повітря на водяному калорифері, що запропонована в [14]:

$$\begin{cases} T_W \frac{d \Delta \theta_W}{dt} + \Delta \theta_W = k_0 \Delta \theta_{W0} + k_1 \Delta \theta_M + k_2 \Delta G_W; \\ T_M \frac{d \Delta \theta_M}{dt} + \Delta \theta_M = k_3 \Delta \theta_W + k_4 \Delta \theta_A; \\ T_A \frac{d \Delta \theta_A}{dt} + \Delta \theta_A = k_5 \Delta \theta_{A0} + k_6 \Delta \theta_M + k_7 \Delta G_A; \end{cases} \quad (1)$$

$$\text{де } K_W = c_W G_W + \alpha_0 F_0, \quad T_W = \frac{c_W M_W}{K_W}, \quad k_0 = \frac{c_W G_W}{K_W}, \quad k_1 = 1 - k_0, \quad k_2 = \frac{c_W (\theta_{W0} - \theta_W)}{K_W};$$

$$K_M = \alpha_0 F_0 + \alpha_1 F_1, \quad T_M = \frac{c_M M_M}{K_M}, \quad k_3 = \frac{\alpha_0 F_0}{K_M}, \quad k_4 = 1 - k_3; \quad K_A = c_A G_A + \alpha_1 F_1, \quad T_A = \frac{c_A M_A}{K_A},$$

$$k_5 = \frac{c_A G_A}{K_A}, \quad k_6 = 1 - k_5, \quad k_7 = \frac{c_A (\theta_{A0} - \theta_A)}{K_A}.$$

Математична модель (1) у просторі станів має вигляд

$$\mathbf{X}' = \mathbf{A}\mathbf{X} + \mathbf{B}\mathbf{U}, \quad (2)$$

$$\text{де } \mathbf{X} = \begin{bmatrix} \Delta \theta_A \\ \Delta \theta_M \\ \Delta \theta_W \end{bmatrix}, \quad \mathbf{A} = \begin{bmatrix} -1/T_A & k_6/T_A & 0 \\ k_4/T_M & -1/T_M & k_3/T_M \\ 0 & k_1/T_W & -1/T_W \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} k_5/T_A & k_7/T_A & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & k_0/T_W & k_2/T_W \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} \Delta \theta_{A0} \\ \Delta G_A \\ \Delta \theta_{W0} \\ \Delta G_W \end{bmatrix}.$$

Для моделі теплообміну (2) запропоновано класифікацію її параметрів у вигляді блок-схеми, яку зображено на рис. 1. Теплофізичні значення параметрів для блоків класифікації (1-3) визначено з високою точністю, вони детально розглянуті в [14]. Коефіцієнти тепловіддачі α_0 , α_1 (блок 4) залежить від багатьох факторів і можуть суттєво змінювати своє значення залежно від: температур матеріальних потоків θ_W , θ_A ; витрати матеріальних потоків G_W , G_A ; вологості повітря; конструктивних особливостей поверхні теплообміну та інших факторів. Ці параметри є предметом наукових досліджень в теплотехніці і розраховується з високою точністю на основі експериментальних досліджень [15]. Із зазначених міркувань коефіцієнти тепловіддачі α_0 , α_1 слід віднести до параметричної невизначеності моделі (2), які суттєво залежать від витрат матеріальних потоків G_W , G_A . В роботі [16] розглянуто вплив коефіцієнта тепловіддачі та витрати повітря на чисельне моделювання динамічних процесів теплообміну для електричного калорифера. Окрім цього, на точність розрахунків впливає "точка" основного статичного режиму теплообмінника, в околі якої проводиться лінеаризація математичної моделі (2).

Від коефіцієнтів тепловіддачі α_0 та α_1 залежать параметри T_W , T_M , T_A , $k_0 \dots k_7$ моделей (1) і (2), оскільки коефіцієнти тепловіддачі входять в розрахункові залежності параметрів моделей. Можна зробити висновок, що від точності визначення числових значень коефіцієнтів тепловіддачі α_0 та α_1 залежать числові значення матриць \mathbf{A} та \mathbf{B} моделі (2), що істотно впливає на результати моделювання динамічних процесів теплообміну.

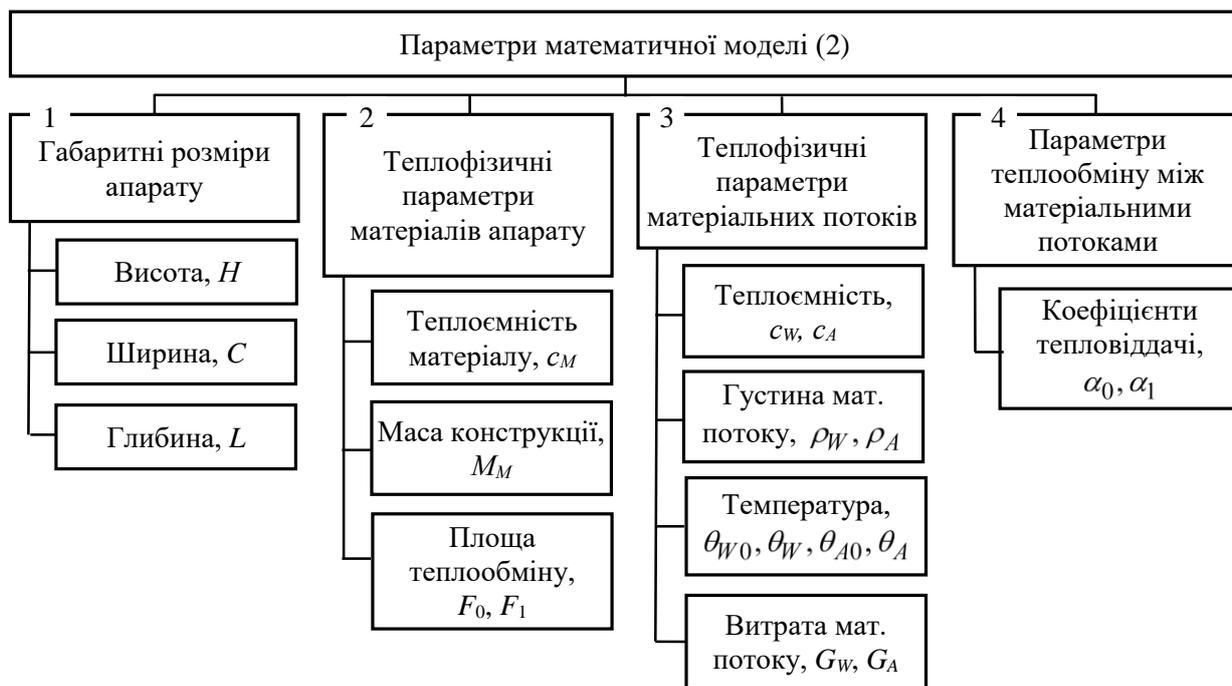


Рис. 1 Класифікація параметрів моделі водяного калорифера

Fig. 1 Classification of the water heater model parameters

3. Ідентифікація параметрів моделі водяного калорифера

Для розкриття концептуальної невизначеності динамічної моделі (2) будемо використати метод пасивного експерименту. Адаптація математичної моделі до даних фізичного процесу, що вимірюються сенсорами КФС, є одним із етапів розробки ЦД. Такий підхід виправданий своєю економічністю, оскільки не потребує додаткових матеріальних ресурсів на виробництві. Використання аналітичних методів адаптації ускладнено нелінійною залежністю параметрів моделі (2) від коефіцієнтів тепловіддачі α_0 та α_1 . З цих міркувань пропонується використати чисельні методи пошуку невизначених параметрів математичної моделі.

Формально динамічна модель (2) містить вісім невизначених параметрів $\alpha_0, \alpha_1, G_W, \theta_{W0}, \theta_W, G_A, \theta_{A0}, \theta_A$. Припустимо, що параметри $\theta_{W0}, \theta_W, \theta_{A0}, \theta_A$ контролюються сенсорами температури. В такому випадку в процесі параметричної ідентифікації моделі (2) необхідно визначити чотири параметри $\alpha_0, \alpha_1, G_W, G_A$. Для пасивної ідентифікації моделі водяного калорифера рекомендується використати методи чисельної оптимізації, оскільки функція пошуку не задана аналітично, а обчислюється безпосередньо під час реалізації алгоритму пошуку. Детально алгоритм пасивної ідентифікації розглянуто у публікації [15].

В алгоритмі ідентифікації застосовується середньоквадратичний критерій мінімізації похибки вимірних змінних стану процесу теплообміну \mathbf{X} і вихідного вектора $\bar{\mathbf{X}}$ оцінок математичної моделі (2), що ідентифікується при вхідному сигналі \mathbf{U}

$$I = M \left\{ \int_{t_0}^{t_0+t_f} (\mathbf{X} - \bar{\mathbf{X}})^T \mathbf{Q} (\mathbf{X} - \bar{\mathbf{X}}) dt \right\} \rightarrow \min, \quad (3)$$

тут t_0 - початковий час тренду, t_f - тривалість дискретизації тренду, \mathbf{Q} - одинична квадратна матриця, \mathbf{T} - оператор транспонування матриці, M - оператор математичного сподівання, який враховує промислові збурення. Узагальнена структурна схема процесу пасивної ідентифікації зображена на рис. 2. Критерій пошуку (3) обчислюється під час реалізації алгоритму ідентифікації.

Метод чисельної оптимізації може мати особливості програмної реалізації. Щоб нівелювати можливість чисельного переповнення критерію (3) необхідно встановити межі пошуку значень параметрів $\bar{\alpha}_0, \bar{\alpha}_1, \bar{G}_W, \bar{G}_A$, виходячи із умови фізичної реалістичності теплообмінного апарату. Інформаційна невизначеність часових трендів нівелюється згладжуванням вимірних змінних для

векторів $U(t)$, $X(t)$, а часовий інтервал трендів t_f повинен бути в декілька разів більшим за тривалість перехідних процесів водяного калорифера, що забезпечує нерівність $t_f > 10 \max(T_W, T_M, T_A)$.

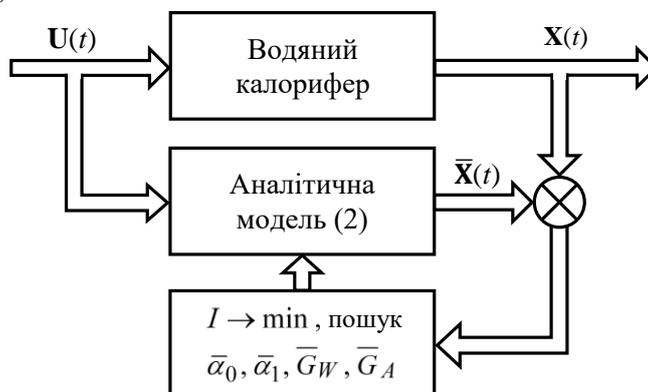


Рис. 2 Структурна схема для ідентифікації моделі водяного калорифера
Fig. 2 Block diagram for identifying the water heater model

Працездатність чисельного методу ідентифікації, що запропонований в роботі [15] оцінено за допомогою програмного пакету MatLAB. У порівнянні із попередніми дослідженнями авторів [16], кількість невизначених параметрів ідентифікації збільшилася, що ускладнило задачу ідентифікації. Якщо для електричного калорифера ідентифікувалися коефіцієнт тепловіддачі та витрата повітряного потоку, то для водяного калорифера ідентифікації підлягають теплофізичні характеристики для двох матеріальних потоків: водяного теплоносія (витрата води G_W та коефіцієнт тепловіддачі α_0); повітря, що нагрівається (витрата повітря G_A та коефіцієнт тепловіддачі α_1).

Для імітаційного моделювання часових трендів $U(t)$ та $X(t)$ робочого водяного калорифера (еталонна модель) застосовувалася модель (2). До векторів еталонних змінних було додано випадковий сигнал з амплітудою ± 0.25 , що відтворює вплив промислових завад у вимірювальних каналах сенсорів. У випадку ідентифікованої моделі (2) початкові значення невизначених параметрів $\bar{\alpha}_0, \bar{\alpha}_1, \bar{G}_W, \bar{G}_A$ суттєво відрізнялися від еталонних значень $\alpha_0, \alpha_1, G_W, G_A$. Для виконання параметричної ідентифікації використано функцію MatLAB `fminsearch(...)`, в якій налаштовано реалізацію методу оптимізації Нелдера-Міда. Основні результати чисельної ідентифікації наведено на рис. 3. На рис. 3 і далі використовуються такі позначення: X вектор стану моделі (2) із еталонними значеннями параметрів $\alpha_0, \alpha_1, G_W, G_A$; вектор стану \bar{X} для моделі (2), що ідентифікується із значеннями коефіцієнтів $\bar{\alpha}_0, \bar{\alpha}_1, \bar{G}_W, \bar{G}_A$, що оптимізуються. Вектори стану X і \bar{X} включають змінні: $\Delta\theta_A$ і $\Delta\bar{\theta}_A$ – температура нагрітого повітря; $\Delta\theta_M$ і $\Delta\bar{\theta}_M$ – температура теплообмінної поверхні; $\Delta\theta_W$ і $\Delta\bar{\theta}_W$ – температура водяного теплоносія. При моделюванні використано теплофізичні характеристики водяного калорифера CVP2-HW2 [16]. Вектор керуючого впливу U математичної моделі (2) містить змінні: $\Delta\theta_{A0}$ – вхідна температура повітря; ΔG_A – витрата повітря; $\Delta\theta_{W0}$ – вхідна температура теплоносія; ΔG_W – витрата водяного теплоносія.

Для ідентифікації параметрів використано випадок, коли початкові умови векторів стану еталонної моделі та моделі, що ідентифікується, співпадають $X(0) = \bar{X}(0) = [1 \ 2 \ 3]^T$, оскільки змінні вектору стану вимірюються. Проте, у дослідженні суттєво відрізняються параметри ідентифікації $\bar{\alpha}_0 = 1200$, $\bar{G}_W = 0.1$, $\bar{\alpha}_1 = 220$, $\bar{G}_A = 0.2$ від еталонних значень $\alpha_0 = 800$, $G_W = 0.15$, $\alpha_1 = 120$, $G_A = 0.3$. При ступінчастій зміні вектора вхідного впливу $U(t) = [1 \ 0.1 \ 1 \ 0.15]^T$, отримано часові характеристики для еталонної моделі X (графіки кривих із випадковою складовою) та ідентифікаційної моделі \bar{X} (графіки пунктирними кривими),

що показані на рис. 3 (а). Різниця між часовими характеристиками простору станів еталонної та ідентифікованої моделі значна.

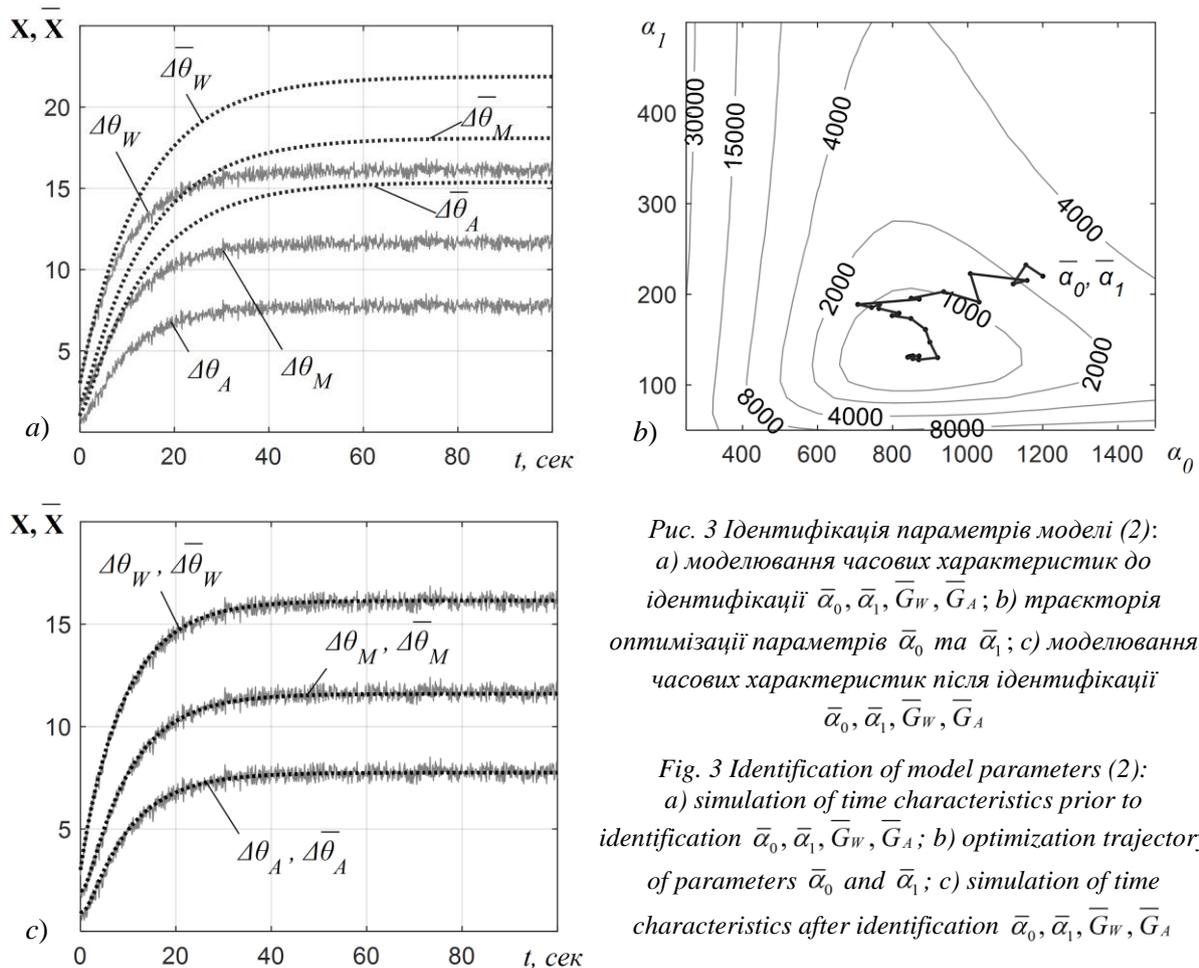


Рис. 3 Ідентифікація параметрів моделі (2): а) моделювання часових характеристик до ідентифікації $\bar{\alpha}_0, \bar{\alpha}_1, \bar{G}_W, \bar{G}_A$; б) траєкторія оптимізації параметрів $\bar{\alpha}_0$ та $\bar{\alpha}_1$; в) моделювання часових характеристик після ідентифікації $\bar{\alpha}_0, \bar{\alpha}_1, \bar{G}_W, \bar{G}_A$

Fig. 3 Identification of model parameters (2): a) simulation of time characteristics prior to identification $\bar{\alpha}_0, \bar{\alpha}_1, \bar{G}_W, \bar{G}_A$; b) optimization trajectory of parameters $\bar{\alpha}_0$ and $\bar{\alpha}_1$; c) simulation of time characteristics after identification $\bar{\alpha}_0, \bar{\alpha}_1, \bar{G}_W, \bar{G}_A$

На рис. 3 (b) показано ізолінії поверхні критерію (3) та траєкторія його мінімізації при умові, що значення параметрів \bar{G}_W, \bar{G}_A уже ідентифіковано. Такий підхід застосовано, оскільки немає можливості графічно продемонструвати оптимізацію критерія (3) за чотирма параметрами. У дослідженні ідентифікувалися усі чотири невизначені параметри при різних початкових умовах. Чисельні дослідження підтвердили, що функціонал (3) має один екстремум в області фізичної реалістичності математичної моделі (2), алгоритм чисельної ідентифікації має добру збіжність при різних початкових умовах та наявності випадкових збурень.

На рис. 3 (c) наведено часові характеристики змінних стану еталонної моделі X та ідентифікованої моделі \bar{X} після ідентифікації невизначених параметрів при вхідній дії $U(t)$ на обидві моделі. Знайдені значення ідентифікованих параметрів є досить близькими до еталонних $\bar{\alpha}_0 = 777.7, \bar{G}_W = 0.15, \bar{\alpha}_1 = 117.7, \bar{G}_A = 0.31$. Відносна похибка ідентифікованих параметрів моделі не перевищила 5%.

4. Розробка цифрового двійника водяного калорифера

Розробку ЦД доцільно здійснювати на основі аналітичної моделі (2). Для адаптації моделі (2) до реального процесу нагрівання повітря водяним калорифером на початковому етапі необхідно виконати ідентифікацію неперервної моделі із застосуванням алгоритму пасивної ідентифікації, розглянутого у попередньому розділі. Подальша реалізація ЦД потребує зменшення обчислювальних ресурсів. Щоб зменшити обчислювальні ресурси на розрахунки моделі ЦД, аналітичну модель (2) необхідно представити у дискретній формі, оскільки на розв'язок різницевого рівняння необхідно менші обчислювальні ресурси ніж на диференціальні рівняння. Розглянемо дискретне представлення неперервної моделі (2) в просторі станів [16]

$$\bar{X}_{k+1} = \bar{A}_d \bar{X}_k + \bar{B}_d U_k, \quad (4)$$

де $\bar{\mathbf{A}}_d = e^{\bar{\mathbf{A}}T_{KV}}$, $\bar{\mathbf{B}}_d = \int_0^{T_{KV}} e^{\bar{\mathbf{A}}(T_{KV}-t)} \bar{\mathbf{B}} dt$, T_{KV} - період квантування. Для перетворення моделі (2)

до дискретної форми (4) достатньо скористатися функцією `c2d(...)` MatLAB.

Для розробки ЦД водяного калорифера скористаємося процедурою синтезу, що розглянута в [15]. Запропонована процедура складається з наступних кроків:

- 1) ідентифікація невизначених параметрів $(\bar{\alpha}_0, \bar{\alpha}_1, \bar{G}_W, \bar{G}_A)$ для математичної моделі (2) за алгоритмом пасивної ідентифікації;
- 2) перехід від неперервної моделі (2) до дискретної моделі (4), яка є ЦД;
- 3) якщо під час роботи точність ЦД погіршилася (через нестационарність фізичного процесу), то переходимо до кроку 1 для уточнення параметрів моделі водяного калорифера.

Імітаційне дослідження розробки та роботи ЦД водяного калорифера виконано в програмному пакеті MatLAB, який використовувався для розрахунку матриць аналітичної моделі (2) та ЦД (4). Результати моделювання наведено на рис. 4.

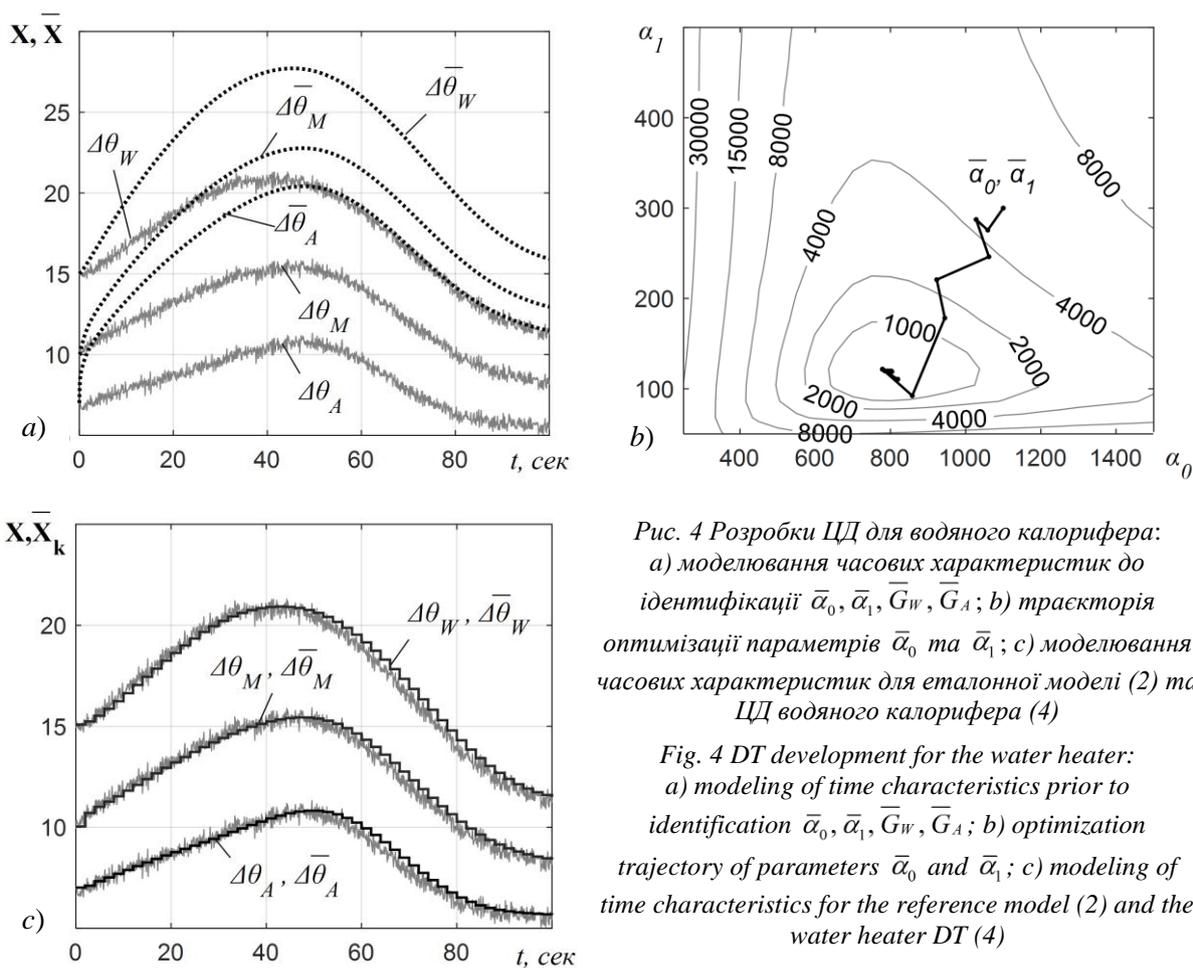


Рис. 4 Розробки ЦД для водяного калорифера: а) моделювання часових характеристик до ідентифікації $\bar{\alpha}_0, \bar{\alpha}_1, \bar{G}_W, \bar{G}_A$; б) траєкторія оптимізації параметрів $\bar{\alpha}_0$ та $\bar{\alpha}_1$; в) моделювання часових характеристик для еталонної моделі (2) та ЦД водяного калорифера (4)

Fig. 4 DT development for the water heater: а) modeling of time characteristics prior to identification $\bar{\alpha}_0, \bar{\alpha}_1, \bar{G}_W, \bar{G}_A$; б) optimization trajectory of parameters $\bar{\alpha}_0$ and $\bar{\alpha}_1$; в) modeling of time characteristics for the reference model (2) and the water heater DT (4)

На рис. 4 (а) наведено приклад симуляції для заданого вхідною впливу $\mathbf{U}(t) = [1 + 0.5 \sin(0.15t) \quad 0.1 + 0.025 \sin(0.1t) \quad 1 + 0.05 \sin(0.3t) \quad 0.15 + 0.05 \sin(0.05t)]^T$ та відповідних початкових умов фізичної моделі $\mathbf{X}(0) = [7 \quad 10 \quad 15]^T$, $\alpha_0 = 800$, $G_W = 0.15$, $\alpha_1 = 120$, $G_A = 0.3$, а також ідентифікованої моделі $\bar{\mathbf{X}}(0) = [7 \quad 10 \quad 15]^T$, $\bar{\alpha}_0 = 1100$, $\bar{G}_W = 0.1$, $\bar{\alpha}_1 = 300$, $\bar{G}_A = 0.2$. На рис. 4 (б) представлено ізолінії поверхні критерію (3) та траєкторію його мінімізації в координатах параметрів α_0 та α_1 , в результаті чисельної ідентифікації знайдено $\bar{\alpha}_0 = 794.4$, $\bar{G}_W = 0.152$, $\bar{\alpha}_1 = 118.9$, $\bar{G}_A = 0.3$. Відносна похибка ідентифікованих параметрів не перевищила 3%. Після параметричної ідентифікації моделі (2) розраховано числові значення матриць ЦД моделі (4) для періоду дискретизації $T_{KV} = 2$:

$$\bar{\mathbf{A}}_d = \begin{bmatrix} 0.0254 & 0.2215 & 0.3130 \\ 0.0323 & 0.2824 & 0.4518 \\ 0.0086 & 0.0852 & 0.7258 \end{bmatrix}; \bar{\mathbf{B}}_d = \begin{bmatrix} 0.4043 & -12.0798 & 0.0358 & 4.7011 \\ 0.1761 & -5.2620 & 0.0573 & 7.5293 \\ 0.0171 & -0.5105 & 0.1633 & 21.4392 \end{bmatrix}.$$

На рис. 4 (с) показано часові характеристики змінних стану для еталонної моделі \mathbf{X} та ЦД $\bar{\mathbf{X}}_k$ водяного калорифера. За результатами імітаційного моделювання можна зробити висновок, що запропонований в роботі [15] підхід до розробки ЦД можна використовувати для складніших аналітичних моделей в просторі станів, що мають чотири невизначених параметри.

Висновки

Модернізація технологічних процесів передбачає інтенсифікацію управління системи теплоспоживання та теплогенерації на промислових підприємствах в рамках КФС виробництва. Глибинний аналіз фізичних процесів можливий в їх ретроспективі із використанням інструментарію ЦД. Впровадження ЦД неминуче призводить до багатовимірності динамічних моделей з невизначеними параметрами, які потребують корекції до конкретних умов застосування.

На основі методології системного аналізу розглянуто процедуру розробки моделі ЦД водяного калорифера із використанням аналітичної моделі в умовах концептуальної невизначеності. Розглянутий підхід розробки ЦД придатний для використання із різними типами теплообмінних апаратів, які широко використовуються в системах теплоспоживання та теплогенерації. В роботі проведено аналіз параметрів аналітичної моделі водяного калорифера. Для аналітичної моделі встановлено наявність чотирьох ключових коефіцієнтів $\bar{a}_0, \bar{a}_1, \bar{G}_W, \bar{G}_A$, які необхідно уточнювати в процесі пасивної ідентифікації та продемонстровано їх вплив на кінцевий результат моделювання. Наведено чисельний приклад алгоритму ідентифікації коефіцієнтів $\bar{a}_0, \bar{a}_1, \bar{G}_W, \bar{G}_A$ для аналітичної моделі водяного калорифера. Показано, що розглянутий алгоритм ідентифікації відноситься до однокрихітальних задач оптимізації. Для чисельних розрахунків використовувалися вбудовані в MatLAB оптимізаційні функції та функції розв'язку диференціальних і різницевих рівнянь. Використання чисельних методів для оцінювання невизначених параметрів аналітичних моделей на основі максимального наближення до реальних даних відкриває нові дослідницькі горизонти.

СПИСОК ЛІТЕРАТУРИ

1. Данилишин Б. Якими мають бути стратегічні напрями відбудови національної економіки. *Економічна правда*. 2022. URL: <https://www.epravda.com.ua/columns/2022/09/1/691022/> (дата звернення: 18.12.2025).
2. Broo D., Boman U., Törngren M. Cyber-physical systems research and education in 2030: Scenarios and strategies. *Journal of Industrial Information Integration*. 2021. Volume 21. 100192. <https://doi.org/10.1016/j.jii.2020.100192>
3. Edward A. Lee Cyber-Physical Systems- Are Computing Foundations Adequate? *NSF Workshop On Cyber-Physical Systems: Research Motivation, Techniques and Roadmap*. October 2006. URL: <https://ptolemy.berkeley.edu/publications/papers/06/CPSPositionPaper/>
4. Government Trends 2024. A report by Deloitte Center for Government Insights. *Deloitte*. URL: https://www2.deloitte.com/content/dam/insights/articles/us187225_gov-trends-24/DI_Gov-trends-24.pdf (дата звернення: 18.12.2025)
5. Pankratova N., Golinko I., Pankratov V. Reliable operation of cyber-physical system with accompanied by a digital twin. *Problems of applied mathematics and mathematical modeling*. 2023. Issue 23. pp. 212-223. <https://doi.org/10.15421/322322>
6. De Benedictis A., Flammini F., Mazzocca N., Somma A., Vitale F. Digital Twins for Anomaly Detection in the Industrial Internet of Things: Conceptual Architecture and Proof-of-Concept. *IEEE Transactions on Industrial Informatics*. 2023. Volume 19, Issue 12. pp. 11553–11563. <https://doi.org/10.1109/TII.2023.3246983>

7. Щеглов В. Р., Морозова О. І. Методи та технології розроблення цифрових двійників для гарантоздатних систем індустріального інтернету речей. *Системи управління, навігації та зв'язку*. 2022. № 4. С. 127–137.
8. Fuller A., Fan Z., Day C., Barlow C. Digital twin: Enabling technologies, challenges and open research. *IEEE Access*. 2020, Volume 8. pp. 108952–108971. <https://doi.org/10.1109/ACCESS.2020.2998358>
9. Pankratova N., Grishyn K., Barilko V. Digital Twins: Stages of Concept Development, Areas of Use, Prospects. *System Research & Information Technologies*. 2023, № 2, pp. 7 – 21. <https://doi.org/10.20535/SRIT.2308-8893.2023.2.01>
10. The Industrial Internet Reference Architecture. *An Industry IoT Consortium Foundational Document*, 2022. URL: <https://www.iiconsortium.org/wp-content/uploads/sites/2/2022/11/IIRA-v1.10.pdf> (дата звернення: 18.12.2025)
11. Pankratova N., Golinko I. Development of Digital Twins to Support the Functioning of Cyber-physical Systems. *Computer Science Journal of Moldova*. 2023. № 3(93), pp. 299–320. <https://doi.org/10.56415/csjm.v31.15>
12. Solovchuk K. Mathematical models for typical continued computer-oriented process control. *Control systems and computers*. 2018. № 5. pp. 79–92. <https://doi.org/10.15407/usim.2018.05.079>
13. Zgurovsky M., Pankratova N. System analysis: Theory and Applications. Springer, 2007. <https://doi.org/10.1007/978-3-540-48880-4>
14. Голінко І., Галицька І. Динамічна модель теплообміну для водяного калорифера у просторі станів. *Інформаційні системи, механіка та керування*. 2016. № 15, С. 83-93. <https://doi.org/10.20535/2219-380415201686207>
15. Pankratova N., Golinko I. Approach to development of digital twin model for cyber-physical system in conditions of conceptual uncertainty. In *Book Chapter M. Zgurovsky, & N. Pankratova (Eds.), System Analysis and Artificial Intelligence (Ser. Studies in Computational Intelligence, 2023. Volume 1107)*. Springer. pp. 3 – 25. https://doi.org/10.1007/978-3-031-37450-0_1
16. Pankratova N., Golinko I. Electric heater mathematical model for cyber-physical systems. *System research & Information technologies*. 2021. №2. pp. 7-17. <https://doi.org/10.20535/SRIT.2308-8893.2021.2.01>

REFERENCES

1. B. Danylyshyn, “What should be the strategic directions of rebuilding the national economy”. *Economic Truth*. 2022. [in Ukrainian] <https://www.epravda.com.ua/columns/2022/09/1/691022/>
2. D. Broo, U. Boman, M. Törngren, “Cyber-physical systems research and education in 2030: Scenarios and strategies”. *Journal of Industrial Information Integration*, Vol. 21, March 2021, 100192. URL: <https://doi.org/10.1016/j.jii.2020.100192>
3. Edward A. Lee, “Cyber-Physical Systems- Are Computing Foundations Adequate?”. *NSF Workshop On Cyber-Physical Systems: Research Motivation, Techniques and Roadmap*, October 2006. <https://ptolemy.berkeley.edu/publications/papers/06/CPSPPositionPaper/>
4. “Government Trends 2024. A report by Deloitte Center for Government Insights”. *Deloitte*. https://www2.deloitte.com/content/dam/insights/articles/us187225_gov-trends-24/DI_Gov-trends-24.pdf
5. N. Pankratova, I. Golinko, V. Pankratov, “Reliable operation of cyber-physical system with accompanied by a digital twin”. *Problems of applied mathematics and mathematical modeling*. 2023. Issue 23. pp. 212-223. <https://doi.org/10.15421/322322>
6. A. De Benedictis; F. Flammini; N. Mazzocca; A. Somma, F. Vitale, “Digital Twins for Anomaly Detection in the Industrial Internet of Things: Conceptual Architecture and Proof-of-Concept”. *IEEE Transactions on Industrial Informatics*, Vol. 19, № 12, December 2023, 11553–11563. URL: <https://doi.org/10.1109/TII.2023.3246983>
7. V. R. Shcheglov, O. I. Morozova, “Methods and technologies for developing digital twins for warrantable systems of the industrial Internet of Things”. *Control, Navigation and Communication Systems*, 2022, No. 4, pp. 127–137. [in Ukrainian]
8. A. Fuller, Z. Fan, C. Day, C. Barlow, “Digital twin: Enabling technologies, challenges and open research”. *IEEE Access*, Volume 8, 2020, pp. 108952–108971.

9. N. Pankratova, K. Grishyn, V. Barilko, “Digital Twins: Stages of Concept Development, Areas of Use, Prospects”. *System Research & Information Technologies*, 2023, № 2, pp. 7 – 21. <https://doi.org/10.20535/SRIT.2308-8893.2023.2.01>
10. “The Industrial Internet Reference Architecture”. *An Industry IoT Consortium Foundational Document*, 2022. <https://www.iiconsortium.org/wp-content/uploads/sites/2/2022/11/IIRA-v1.10.pdf>
11. N. Pankratova; I. Golinko, “Development of Digital Twins to Support the Functioning of Cyber-physical Systems”. *Computer Science Journal of Moldova*, vol.31, № 3(93), 2023, pp. 299–320. <https://doi.org/10.56415/csjm.v31.15>
12. K. Solovchuk, “Mathematical models for typical continued computer-oriented process control”. *Control systems and computers*, 2018, № 5, pp. 79–92. <https://doi.org/10.15407/usim.2018.05.079>
13. M. Zgurovsky, N. Pankratova, “System analysis: Theory and Applications”. Springer, 2007. <https://doi.org/10.1007/978-3-540-48880-4>
14. I. Golinko, I. Galytska, “Dynamic heat transfer model for a water heater in state space”. *Information Systems, Mechanics and Control*, No. 15, 2016, pp. 83-93. [in Ukrainian] <https://doi.org/10.20535/2219-380415201686207>
15. N. Pankratova, I. Golinko, “Approach to development of digital twin model for cyber-physical system in conditions of conceptual uncertainty”. In *Book Chapter M. Zgurovsky, & N. Pankratova (Eds.), System Analysis and Artificial Intelligence*. Springer, 2023, Volume 1107, pp. 3 – 25. https://doi.org/10.1007/978-3-031-37450-0_1
16. N. Pankratova, I. Golinko “Electric heater mathematical model for cyber-physical systems”. *System research & Information technologies*, №2. 2021. pp. 7-17. <https://doi.org/10.20535/SRIT.2308-8893.2021.2.01>

Igor Golinko

Associate Professor, Associate Professor of the Department of Energy Processes Automation, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 37, Prospect Beresteiskyi, Kyiv, 03056

e-mail: golinko.igor@iit.kpi.ua

<https://orcid.org/0000-0002-7640-4760>

Digital Twin Development for the Heat Transfer Process under Conceptual Uncertainty

Relevance. Industrial development is characterized by the active introduction of cyber-physical systems, a key element of which is digital twins. Digital twins make it possible to improve the efficiency of technological process management, ensure the prediction of equipment operating modes, optimize energy consumption, and reduce operating costs. Water heaters are an important component of heat generation and consumption systems in industrial facilities, which necessitates the creation of adequate models capable of accurately reflecting real heat exchange processes. However, analytical models of such equipment often contain uncertain parameters, which reduces the accuracy of modeling and complicates their practical application without additional identification.

Purpose of the publication is to develop a water heater digital twin based on an analytical model adapted to real heat exchange conditions by identifying uncertain parameters. The analytical model is considered as a basic example that can be generalized for different types of heat exchange equipment.

Methods. The work uses analytical modeling of thermal processes and passive identification methods of mathematical model parameters. The dynamic model is adapted by minimizing the quadratic quality criterion, which characterizes the deviation of the state space variables of the mathematical model from the experimental data of the real heat transfer process. Numerical methods are used for passive identification.

Results. An analysis of the analytical mathematical model of a water heater was carried out and four uncertain parameters were identified that need to be refined to ensure the adequacy of the model. These parameters include material flows rates and heat transfer coefficients that determine the intensity of the heat flow through the heat exchange surface of the device. Based on numerical modeling, it is shown that the task of identifying uncertain coefficients is single-extreme in nature, which ensures the stability of optimization results and the possibility of applying standard numerical methods. The results of numerical modeling confirmed the effectiveness of the proposed approach to adapting the analytical model and developing a digital twin of a water heater.

Conclusions. The proposed approach to the digital twin development of a water heater provides an adequate reproduction of the real heat exchange process and can be used as part of industrial enterprises cyber-physical systems. The considered identification method can be easily extended to other types of heat exchange equipment used in heat generation and consumption systems.

Keywords: *cyber-physical system, digital twin, mathematical model, state space, water heater, heat exchange, passive identification.*

УДК (UDC) 004.8

**Костюченко Альбіна
Валентинівна**

магістрант кафедри системного програмування та спеціалізованих комп'ютерних систем, "Київський політехнічний інститут імені Ігоря Сікорського", 03056, Україна, Київ, вул. Політехнічна, 14-а
e-mail: albina.kostyuchenko03@gmail.com
<https://orcid.org/0009-0004-7382-7209>

**Петрашенко Андрій
Васильович**

к.т.н., доцент кафедри системного програмування та спеціалізованих комп'ютерних систем, Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського", 03056, Україна, Київ, вул. Політехнічна, 14-а
e-mail: petrashenko@gmail.com;
<https://orcid.org/0000-0003-0239-1706>

Метод генерації опису програмного коду з використанням моделі штучного інтелекту

Актуальність. Тема є актуальною, оскільки у даний час існує багато великих проектів, які розробляються протягом тривалого періоду часу та потребують підтримки та розуміння коду без пояснень. Швидкий розвиток технологій та необхідність постійної розробки нових функцій і підтримки вже існуючих потребує постійного оновлення документації. Написання хорошої документації є цінною навичкою, що потребує досвіду, концентрації та розуміння структури проекту. Як наслідок, велика кількість розробників вважають процес написання документації важким і думають, що час, витрачений на це, можна було б використати більш продуктивно. Саме тому є попит на сервіси, які допомагають автоматизувати цей процес.

Мета. Метою даної роботи є підвищення ефективності автоматизованої генерації програмної документації. У рамках виконання даного завдання було опрацьовано необхідний теоретичний матеріал, вивчено уже існуючі рішення даної проблеми та розроблено і реалізовано власний новий метод генерування опису програмного коду, який більш точно визначав призначення фрагментів коду, чітко розумів структуру та залежності між його складовими.

Методи дослідження. Дослідження базується на аналізі літератури, статистичних методах, а також методах машинного навчання та інтелектуального аналізу даних. Зокрема було використано методи синтаксичного аналізу коду та побудови абстрактного синтаксичного дерева (AST), метод формування навчального корпусу, методи навчання та донавчання трансформерних та графових моделей. Для оцінки переваг донавченої моделі було застосовано метод порівняльного моделювання та автоматизованої оцінки якості тексту (у даному випадку BERTScore).

Результати. Донавчання моделі T5 на спеціалізованому наборі даних із прокоментованим кодом у поєднанні з лексичним аналізом дозволило підвищити якість генерації приблизно на 4% за метрикою F1 порівняно з базовою моделлю. Це свідчить про те, що адаптація моделі до конкретної доменної задачі є ефективною та здатною суттєво покращити результат.

Висновки. На основі зібраних даних було запропоновано власний підхід покращення якості генерації опису коду з використанням донавченої моделі T5 та створеної моделі GNN з подальшою реалізацією, що і є результатом дослідження. Запропонована система поєднує кращі практики синтаксичного аналізу, графового моделювання і трансформерної генерації, забезпечуючи практично застосовне рішення для автоматичного створення документації. Можна стверджувати, що поєднання «seq2seq» моделей, методів токенизації та адаптації великих трансформерів, а також аналізу коду через GNN і структурні AST-представлення забезпечує комплексний підхід до автоматизації роботи з кодом, дозволяючи поєднувати локальні й глобальні контексти, швидко адаптувати модель під специфічні задачі та ефективно генерувати змістовні коментарі та документацію. Такий інтегрований підхід має потенціал для подальшого розвитку систем штучного інтелекту у сфері автоматичного аналізу коду, підвищення продуктивності розробників та забезпечення якості програмного забезпечення. Результати дослідження можуть бути застосовані на практиці для швидкого та ефективного створення документації до розроблюваного програмного забезпечення та великих проектів мовою Python.

Ключові слова: машинне навчання, T5, GNN, генератор опису коду, навчання моделі, обробка природної мови, документація, AST.

Як цитувати: Костюченко А. В., Петрашенко А. В. Метод генерації опису програмного коду з використанням моделі штучного інтелекту. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління*. 2025. вип. 68. С. 30-42. <https://doi.org/10.26565/2304-6201-2025-68-03>

How to quote: A. Kostiuchenko, and A. Petrashenko, “Method for generating source code description using an artificial intelligence model”, *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 68, pp. 30-42, 2025. <https://doi.org/10.26565/2304-6201-2025-68-03>

Вступ

У наш час сфера інформаційних технологій розвивається надшвидко. На даний момент Github, найбільший у світі веб-сервіс для зберігання та публікації коду, налічує більше 47 мільйонів публічних репозиторіїв. Проте, на жаль, не весь код, що існує, є зрозумілим та має чіткі та розгорнуті пояснення у вигляді коментарів, ще рідше зустрічається повноцінна, чітко структурована документація. І це є досить серйозною проблемою, оскільки над великими проектами часто працює не одна сотня розробників, котрі час від часу змінюються. Відсутність якісної документації підвищує кількість часу, необхідного для розуміння та редагування коду.

Саме тому існує програмне забезпечення, яке генерує документацію на основі коментарів у кодї, назв та функціональності класів та методів або загального контексту. Найстаріша система для документування коду була створена ще у 1997 році, проте стрімко розвиватись ця галузь програмного забезпечення почала менше двадцяти років тому – з появою моделей-трансформерів. Певні програми створюють опис коду за шаблонами, інші використовують обробку природної мови для аналізу залежностей та функцій, які виконують ті чи інші частини коду.

Генератори за шаблонами, такі як Swagger для API або Doxygen для різних мов програмування, працюють за принципом аналізу структури коду та специфікацій і подальшого створення документації у стандартизованому форматі. Вони дозволяють швидко отримати базову документацію: наприклад, список методів, параметрів, типів даних і прості пояснення до функцій. Основні переваги таких систем – це висока швидкість, передбачуваність результату та мінімальна потреба у навчанні чи складних ресурсах. Однак ці підходи мають й істотні обмеження: вони не враховують семантику бізнес-логіки, не генерують інтелектуальні пояснення коду та обмежені тими шаблонами, які передбачені розробником генератора. Тобто документація часто є формальною, поверхневою і не допомагає зрозуміти, що насправді робить конкретний фрагмент коду.

Інший клас систем – це генератори на основі штучного інтелекту, наприклад, GitHub Copilot або Tabnine. Вони використовують великі трансформерні моделі, навчальні на масивних репозиторіях коду, і здатні пропонувати коментарі, завершення коду та навіть цілі функції на основі контексту. Переваги таких рішень очевидні: вони адаптуються до стилю коду, можуть пропонувати нестандартні рішення та інтелектуальні коментарі. Водночас вони мають ряд недоліків: висока обчислювальна складність, залежність від хмарних сервісів та великих навчальних наборів, ризик генерації помилкового або небезпечного коду, а також відсутність контролю над точністю та стилем коментарів. Крім того, такі системи є закритими і не дозволяють повністю кастомізувати процес генерації під конкретну доменну задачу.

Метою даної роботи є підвищення ефективності автоматизованої генерації програмної документації. Метод, застосований у даному проекті, поєднує переваги штучного інтелекту і параметроефективного дообучення, одночасно мінімізуючи недоліки обох попередніх методів. Використання моделі T5, донавченої методом LoRA на наборі фрагментів коду з коментарями, дозволяє генерувати семантично точні та структуровані коментарі, що враховують як синтаксис, так і логіку коду. На відміну від генераторів за шаблонами, створена система не обмежена суворими правилами форматування і здатна створювати інтелектуальні пояснення. На відміну від загальних AI-сервісів, таких як Copilot, модель навчена на специфічному наборі даних, що дозволяє контролювати якість коментарів, адаптувати їх під конкретний стиль або проект, а також повністю працювати локально без залежності від сторонніх хмарних сервісів.

1. Існуючі рішення за темою

Найвідомішим генератором документації на даний момент є Doxygen – це система документування вихідних текстів, яка була вперше випущена 26 жовтня 1997 року і на даний момент підтримує десять мов програмування, серед яких PHP, Java, C, Fortran і VHDL. Система підтримує генерацію документації у форматах HTML, XML, LaTeX, RTF та man. Doxygen досить широко застосовується і використовується у таких проектах як Torque Game Engine, AbiWord, Mozilla, Crystal Space та інших.

У першу чергу програма аналізує файл конфігурації, котрий керує налаштуваннями проекту. Отримані налаштування зберігаються у файлі `src/config.h` у класі `Config`. Сам парсер, написаний за допомогою `flex` (Fast Lexical Analyzer Generator) – генератора лексичних аналізаторів, розбиває текст на окремі токени за допомогою вказаних регулярних виразів. Парсер знаходиться у окремій бібліотеці `src/config.l`.

Кожен параметр конфігурації має тип `String`, `List`, `Enum`, `Int` або `Bool`. Їхні значення глобально доступні через функції типу `Config_getXXX()`, де `XXX` – тип параметра. Наприклад, функція `Config_getBool` (`GENERATE_TESTLIST`) повертає вказівник на булеве значення `True`, якщо `testlist` доступний у файлі конфігурації.

Далі всі файли з вхідним кодом, які були вказані у конфігураційному файлі, за замовчуванням проходять обробку спеціалізованим препроцесором `C`, який створений за допомогою того ж `flex` і має кілька відмінностей від стандартного та знаходиться у файлі `src/pre.l`. По-перше, препроцесор за замовчуванням не розширює виклики макросів, хоча це можна налаштувати, по-друге – команди `#include` аналізуються, але вказані там файли не підключаються. Кожен файл викликається функцією `preprocessFile()`, а результат попередньої обробки додається до буферу символів у форматі «`0x06 file name N; 0x06 preprocessed contents of file N`».

Отримані файли обробляє аналізатор мови (файл `src/scanner.l`), який працює як кінцевий автомат з використанням `flex`. Задача аналізатора – це перетворення отриманого на попередньому кроці буфера символів у абстрактне синтаксичне дерево. Для всіх мов працює один великий сканер, що сповільнює роботу програми.

Після цього створюються словники вилучених класів, файлів, просторів імен, змінних, функцій, пакетів, сторінок і груп. Окрім створення словників, під час цього кроку обчислюються відносини (наприклад, відносини спадкування) між отриманими сутностями. Наявні у коді блоки коментарів зберігаються як рядки в сутностях, які вони документують.

Якщо увімкнено перегляд вихідного коду або якщо в документації зустрічаються фрагменти коду, запускається аналізатор вихідного коду. Синтаксичний аналізатор коду намагається зробити перехресне посилання на вихідний код, який він аналізує, із задокументованими сутностями. Він також виконує підсвічування синтаксису джерел. Вихідні дані записуються безпосередньо до вихідних генераторів. Основна точка входу для аналізатора коду `parseCode()` оголошена в `src/code.h`.

Після збору даних і перехресних посилань програма оформлює вихідні дані у різних форматах. Для цього використовуються методи, надані абстрактним класом `OutputGenerator`. XML-дані генеруються безпосередньо із зібраних структур даних. У майбутньому розробники планують використовувати XML як проміжну мову. Перевага наявності проміжної мови у тому, що незалежно розроблені інструменти, написані різними мовами, зможуть витягувати інформацію з вихідних даних XML.

Ще одним відомим інструментом, який у певному сенсі став стандартом для веб-розробки, є `Swagger` – інструмент для створення документації до API-сервісів. Він створює окрему сторінку з інтерактивною документацією, де відповідно до шаблону описано кожен із задокументованих функцій з прикладами використання та зразками вхідних даних, які функція отримує або створює. Такий підхід дозволяє зручно протестувати розроблений API одразу у браузері.

Інструкції для `Swagger` можна записувати у форматі `JSON` або `YAML`. Структура опису API відповідає стандартам `OAS` (`OpenAPI Specification`). Зокрема можна вказати специфікацію `Swagger`, назву, короткий опис та версію створеного API, окремі кінцеві точки (шляхи) та методи (операції) HTTP, які підтримуються цими кінцевими точками, методи аутентифікації, тощо.

Серверні елементи `Swagger`, такі як `Swagger Codegen`, `Swagger Parser`, написані мовою `Java`, інтерактивна документація `Swagger UI` створюється за допомогою мов `Javascript`, `HTML` та `CSS`. `Swagger Editor`, який дозволяє перевіряти специфікації на відповідність стандартам `OpenAPI Specification` та тестувати API у браузері, створений за допомогою `ReactJS`. Також для різних мов

та фреймворків існують бібліотеки Swagger Annotations, Flask-Swagger, FastAPI та інші, які дозволяють додавати анотації у програмний код.

Далі анотації після валідації автоматично перетворюються у OAS-файл. Swagger-сервери аналізують вхідний код, використовуючи статичний аналіз та рефлексію, щоб отримати інформацію про API. Потім всі зібрані дані перетворюються у інтерактивний односторінковий додаток, що складається з шаблонів. При цьому використовуються бібліотеки для рендерингу, такі як Handlebars.

2. Огляд досліджень та публікацій за темою

На одному з перших етапів дослідження було опрацьовано ряд публікацій, які так чи інакше розглядають використання моделей штучного інтелекту.

Зокрема у статті “Automatic comment generation for source code using external information by neural networks for computational thinking” автори пропонують підхід до автоматичної генерації коментарів до вихідного коду, орієнтований на освітнє середовище та розвиток «комп’ютерного мислення». У статті вирішуються декілька фундаментальних проблем, що пов’язані із розумінням коду та побудовою зрозумілих коментарів, і для цього пропонуються конкретні архітектурні рішення. Дана робота є важливим дослідженням у галузі обробки коду нейронними мережами, оскільки вона розглядає не лише сам код, але й додатковий контекст, який супроводжує програму, що є суттєво новим.

Першою ключовою проблемою, на яку звертають увагу автори, є недостатність інформації, що міститься безпосередньо у вихідному коді. У багатьох випадках синтаксис або структура програми не дозволяють однозначно визначити її призначення, і без додаткового контексту модель не здатна коректно інтерпретувати логіку роботи. Для подолання цього обмеження дослідники вводять концепцію зовнішньої інформації, що включає формулювання задачі, опис вхідних і вихідних даних, текст навчального завдання та інші пояснювальні матеріали. Цей текст конкатенується з програмним кодом і подається на вхід моделі типу encoder–decoder на базі LSTM. Таким чином, модель працює не лише з абстрактними синтаксичними структурами, а й з семантичним описом, що значно підвищує релевантність і змістовність згенерованих коментарів.

Автори багатьох попередніх робіт створювали програми, що генерують описи на рівні всієї функції чи файлу, однак іноді необхідно пояснювати роботу програми поетапно, на рівні окремих рядків або логічних блоків. Тому автори роблять акцент на деталізації коментарів і розробляють метод розбиття коду на структурні фрагменти – цикли, умовні оператори, декларації змінних та інші елементи. Для кожного такого блоку формується пара «код–коментар», що дозволяє навчати модель точковому поясненню дій програми, а не створенню узагальнених описів. Такий підхід робить результати моделі особливо корисними для початківців, оскільки пояснення стають локальними й прив’язаними до конкретних частин коду.

Ще однією задачею, яку автори вирішують у своїй роботі, є оцінювання якості згенерованих коментарів. Оскільки автоматичні метрики мають обмеження, дослідники використовують комбінований підхід. Для статистичного порівняння було застосовано метрики BLEU, ROUGE та METEOR, а для суб’єктивної оцінки зрозумілості, правильності та корисності отриманих пояснень залучили студентів і викладачів. Така двошарова система оцінювання забезпечує більш повне розуміння ефективності моделі в реальних освітніх сценаріях.

У технічному плані модель побудована на класичній архітектурі seq2seq з використанням LSTM в ролі енкодера та декодера, доповненої механізмом уваги. Це дозволяє декодеру фокусуватися на різних частинах коду під час генерації коментаря. Навчання моделі здійснюється за допомогою функції Cross-Entropy Loss, що є стандартним підходом у задачах генерування тексту. Об’єднання коду і зовнішнього тексту в один послідовний вхід дозволяє моделі будувати репрезентації, які враховують як синтаксичну, так і семантичну інформацію.

Таким чином, дана робота вирішує одразу кілька ключових проблем: нестачу семантичного контексту, відсутність деталізованих описів окремих блоків коду та складність об’єктивного оцінювання якості згенерованих коментарів, обмежуючись лише автоматичним аналізом і метриками.

Хоча запропонований підхід не враховує структурні залежності між модулями та не використовує абстрактні синтаксичні дерева чи графові моделі, він задає концептуальну основу, на якій можна будувати більш складні системи. У контексті даної магістерської роботи ця публікація служить важливим джерелом для порівняння, оскільки запропонована в дослідженні

система може бути розширена за рахунок використання AST, графових нейронних мереж та сучасних моделей трансформерного типу, що дозволяє досягти вищої точності та структурної узгодженості згенерованої документації.

Наступним опрацьованим дослідженням є “Retrieve and Refine: Exemplar-based Neural Comment Generation”, у якому автори пропонують інноваційний підхід до автоматичної генерації коментарів для коду, який поєднує нейронні трансформерні моделі із механізмами пошуку найбільш релевантних прикладів у великій базі коментованого коду. На відміну від традиційних seq2seq-моделей, які генерують коментар безпосередньо з програмного коду, автори використовують двоступеневу архітектуру типу retrieve-and-refine, що дозволяє моделі використовувати вже існуючі якісні коментарі як відправну точку для побудови нового тексту.

Автори стверджують, що моделі мають обмежені можливості щодо генерації осмислених коментарів лише на основі вихідного коду. Особливо це помітно, якщо програма містить складні логічні конструкції, або використовує домену термінологію, яку модель без прикладів іноді інтерпретує неправильно. Для подолання цього обмеження дослідники впроваджують компонент retrieve, який за допомогою подібності коду знаходить приклад – існуючий фрагмент коду з якісним коментарем. Таким чином, система отримує не лише сам код, а й коментар, створений спеціалістом для схожої структури або логічного шаблону.

Другою важливою складовою системи є компонент refine, який дозволяє трансформеру змінювати, адаптувати та редагувати знайдений коментар під новий фрагмент коду. Автори застосовують модифіковану архітектуру типу енкодер–декодер, де енкодер отримує як сам код, так і коментар-прецедент. Декодер, у свою чергу, генерує новий коментар, поєднуючи інформацію з обох джерел. Тобто модель не створює текст з нуля, а відштовхується від уже наявного людського прикладу, що суттєво підвищує якість, змістовність та стильову природність згенерованих коментарів.

Третьою проблемою, на яку звертають увагу автори, є недостатня структурна адекватність коментарів, згенерованих чистими нейронними моделями. Моделі без прикладів часто дають надто загальні, неконкретні або неправильні пояснення. Підхід retrieve-and-refine використовує механізм стилістичного та семантичного наслідування: знайдений приклад надає релевантну лексику, структуру пояснення та термінологію. Завдяки цьому вихідний коментар має більше шансів бути зрозумілим, технічно точним і близьким до того, що пишуть реальні розробники.

Ще однією важливою частиною роботи є методика оцінювання моделі. Автори застосовують стандартні текстові метрики (BLEU, ROUGE), але також тестують модель на семантичних метриках, які краще відображають відповідність змісту, а не лише поверхневу схожість токенів. Крім того, вони проводять людську оцінку коментарів із залученням програмістів, які оцінюють такі параметри, як коректність, інформативність та відповідність коду. Результати демонструють, що підхід retrieve-and-refine суттєво перевершує моделі, які генерують текст без доступу до прецедентів.

З технічної точки зору архітектура моделі передбачає окремий етап обчислення схожості коду, для якого можуть використовуватися багатовимірні векторні представлення, ембеддинги або моделі з попереднім навчанням. Після того як найбільш схожі приклади відібрано, нейронна модель, заснована на трансформері, здійснює подальше уточнення, створюючи фінальний коментар. Навчання моделі виконується за допомогою крос-ентропійної функції втрат (Cross-Entropy Loss), що дозволяє покращити якість тексту на виході.

Що стосується графових нейронних мереж, ідея формування дерева проекту і його збереження у вигляді графа не є новою. Зокрема у доповіді «JGNN: Graph Neural Networks on Native Java» автор описує, як перетворити програмний проект у граф залежностей. Використовуються зв'язки імпорту, виклики функцій, структуру пакета та результати аналізу AST. Отриманий у результаті дослідження граф орієнтований і неоднорідний, оскільки містить кілька типів вузлів і зв'язків.

Розглянуто різні архітектури графованих нейронних мереж (GCN, GraphSAGE, GAT), націлювання ребер, детально пояснено механізм передачі повідомлень - властивість GNN, що дозволяє вузлу збирати інформацію від сусідів і розглядати контекст всього проекту.

Проте метою проекту було не документування коду, а пошук вразливостей. Створена модель навчена поділяти вузли графа на вразливі та безпечні. Було важливо, щоб GNN враховувала не тільки локальні характеристики файлу, але і його оточення: наявність небезпечних залежностей,

з'єднань, маршрутів поширення даних. Це особливо актуально для сучасних проєктів з великою кількістю зовнішніх бібліотек.

У статті показано, що класичні інструменти обмежуються лінійним або модульним аналізом коду, у той час як графові методи фіксують залежності від файлу до файлу глобально. Автор наводить приклади вразливостей, які точно визначені через інтермодульний контекст. З цією метою формується група програмних проєктів з помітними вразливостями. Вилучення залежностей і побудова графу залежностей відіграє важливу роль, що вимагає попереднього аналізу AST і метаданих проєкту, таких як `package.json`, `requirement.txt` тощо.

3. Теоретичні відомості щодо основних складових системи

3.1. Особливості структури та навчання моделі T5

Модель Text-to-Text Transfer Transformer, також відома як T5, була створена Google Research у рамках проєкту "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer". T5 використовується для обробки природної мови і використовує формат `text-to-text`, тобто вхідні і вихідні дані представлені у вигляді тексту, що спрощує роботу з моделлю. На даний момент існує кілька конфігурацій моделі, зокрема Small, що містить близько 60M параметрів, Base (близько 220 мільйонів), Large (близько 770 мільйонів), 3B (3 мільярдів) і 11B (більше 11 мільярдів параметрів).

Робочий процес T5 складається з двох етапів: попереднього навчання (`pre-training`) та тонкого налаштування (`fine-tuning`). Під час попереднього навчання модель навчається на величезному корпусі загальнодоступних текстових даних. Стандартна T5 навчена на корпусі C4, який складається з 745 ГБ фільтрованих текстів, очищених від нерелевантних даних, частин коду та дублікатів. Метою навчання є передбачення замаскованих токенів у вхідному тексті, подібно до популярної моделі BERT. Однак, на відміну від BERT, де дані кодуються в числа, T5 використовує підхід тексту в текст, де і вхід, і вихід є текстовими представленнями.

Тонке налаштування тренує модель під конкретну задачу, надаючи завдання в текстовому вигляді, наприклад, `«summarize: [text]»`. Наприклад, існує модифікація CodeT5, навчена на масивах коду та натренована для вирішення задач розробки програмного забезпечення.

Для виконання задач машинного навчання, особливо глибоких нейронних мереж, компанія Google розробила TPUs (Tensor Processing Units) – спеціальні прискорювачі, кожен з яких складається з великої кількості ядер, кожен з яких виконує обчислення в форматі тензорів і використовує принципи паралельності даних та паралельності моделі. TPUs використовуються у тому числі і у моделі T5. Окрім цього, використовується алгоритм Adafactor – оптимізатор, який доповнює TPU і дозволяє більш ефективно використовувати пам'ять, що особливо актуально при роботі з великими моделями та за умов браку ресурсів.

Для донавчання моделі було використано метод LoRA, який дозволяє адаптувати великі трансформери, вводячи додаткові матриці низького рангу до ваг ключових шарів моделі. Зазвичай це роблять для ваг `query` та `value` у механізмі `attention`. Особливостями методу є те, що основні параметри моделі заморожуються, натомість додаються менші адаптерні матриці `A` та `B` (1). Вихід кожного шару змінюється таким чином:

$$W' = W + \Delta W = W + A \cdot B \quad (1)$$

де W – незмінні ваги оригінальної моделі, ΔW – навчувані матриці низького рангу.

Граденти обчислюються лише для цих матриць, а решта моделі лишається незмінною. Це дозволяє моделі швидко підлаштуватися під нову доменну інформацію при дуже малому додатковому обсязі пам'яті.

Такий метод може трохи поступатися у точності повному `fine-tuning` і не завжди підходить для завдань, де потрібна глибока модифікація всіх шарів моделі. Проте алгоритм LoRA компенсує свої недоліки тим, що у разі зменшує потребу в пам'яті у процесі навчання та забезпечує швидке тренування, оскільки оновлюються лише додаткові параметри. Такий метод підходить навіть для великих моделей.

Метод LoRA часто застосовують для адаптації моделі на невеликому наборі даних та загалом для донавчання для вузьконаправленої задачі. Наприклад, для створення спеціалізованих чат-ботів або донавчання під специфічний стиль тексту.

У даному випадку було доцільним використати саме цей підхід, адже наявні ресурси були обмеженими і створення моделі з нуля або навіть повний `fine-tuning` були б складними і не дали

бажаного результату. Модель T5-base вже навчена на великому корпусі очищених даних і ефективно працює з текстом, тому часткова її адаптація для задач коментування коду можна вважати раціональним з точки зору використання ресурсів. Завдяки замороженню основних параметрів T5 зберігається знання про загальні мовні закономірності та семантику коду, що дозволяє моделі генерувати коментарі більш природно та інформативно.

3.2. Особливості структури та навчання моделей типу GNN

У рамках цієї роботи була розроблена та навчена графова нейронна мережа (GNN) для аналізу залежностей між файлами та кодovими модулями з метою побудови діаграм імпорту та експорту.

Представлення коду формується таким чином: кожен файл або модуль розглядається як вузол графа, а ребра між вузлами відповідають залежностям, таким як імпорт, виклики функцій або посилання на класи. Кожне з'єднання може бути додатково закодовано з характеристиками, що відображають його тип і важливість. Кожен вузол має вкладення, які відображають його характеристики: наявність docstring, кількість функцій і класів, довжину коду і частоту виклику функцій. Ребра кодуються за допомогою двійкових індикаторів типів залежностей, що дає змогу розрізняти ключові імпорти та вторинні посилання.

Для архітектури моделі було обрано багатопшарову Graph Attention Network (GAT), що складається з чотирьох шарів, кожен з яких агрегує інформацію від сусідніх вузлів за допомогою механізму уваги. Такий підхід дозволяє диференціювати важливість різних залежностей, забезпечуючи більш точне представлення структури проекту. оскільки він дозволяє призначити різні ваги сусіднім вузлам при агрегуванні інформації. Це особливо важливо в задачах аналізу коду, де деякі імпорти або ключові функції мають набагато більший вплив на структуру та поведінку програми.

Послідовність шарів GAT виконують агрегацію ознак сусідніх вузлів з подальшим застосуванням нелінійної функції активації ReLU. Між шарами використовується нормалізація BatchNorm і регуляризація, що забезпечує стабільність навчання і запобігає перенавчанню. На вихідному рівні модель генерує вкладення вузлів, які потім використовуються для побудови візуальної діаграми залежностей.

Навчання моделі проводилося за допомогою оптимізатора AdamW з швидкістю навчання $1e-3$ і зниженням ваги $1e-5$. Cross-Entropy Loss використовувався як функція втрат для класифікації вузлів за важливістю та ідентифікації ключових залежностей. Щоб запобігти перетренованості, на вузли та ребра наносили відсівання 0, 1. Для великих графів застосовувалося пакетування з `torch_geometric.attn_loader`, що дозволило ефективно тренувати модель на декількох файлах і сховищах одночасно.

Під час навчання були обрані оптимальні параметри: чотири шари GAT, 8 голівок уваги в кожному шарі, розмір вбудовувань вузлів 128 і відсівання 0, 1. Ця конфігурація дозволила моделям точно визначити ключові залежності, включаючи важливі імпорти і функції, а також сформувати схему проекту в графік, придатний для візуалізації та подальшої генерації документації. В результаті навчений GNN забезпечує високу точність передбачень залежностей і є ефективним інструментом для побудови структурованих схем коду, інтегрованих в систему автоматичної генерації документів, що використовується в цій роботі.

3.3. Роль ембедингів у запропонованому методі

Абстрактне синтаксичне дерево (AST) є ключовим представленням програмного коду, яке відображає структуру програми у вигляді ієрархії вузлів – операторів, функцій, класів, імпортів, викликів методів тощо. Однак саме по собі AST є символічним, а тому не може бути безпосередньо використане нейронними мережами. Для інтеграції цієї структурної інформації в GNN необхідно перетворити кожен вузол AST на числовий вектор. Такі вектори називаються AST Node Embeddings.

AST Node Embedding – це компактне векторне представлення вузла AST, яке кодує як його тип (наприклад, `FunctionDef`, `Assign`, `ClassDef`), так і його внутрішній зміст (імена змінних, ключові слова, літерали) та структурні характеристики – глибина у дереві, кількість дочірніх елементів, розмір блоку коду, тощо. Формування таких ембедингів включає декілька етапів.

Спочатку, за допомогою ANTLR або стандартних засобів Python, вихідний код перетворюється на AST. Кожен вузол дерева класифікується за типом, а тип вузла проходить

через вбудований embedding-шар, який навчається подібно до словникових embedding у NLP. Це дозволяє моделі зрозуміти, що певні конструкції мови програмування мають схожі ролі.

Далі з вузла виділяється семантичний зміст: імена змінних, назви функцій, значення літералів або ключові слова. Ці токени обробляються за допомогою токенизатора, після чого їхні векторні представлення агрегуються – наприклад, через середнє або згортку. Таким чином модель отримує інформацію про текстовий зміст вузла.

Окрім цього, формуються числові ознаки, що описують структуру: кількість дочірніх вузлів, довжина тіла функції, кількість параметрів, рівень вкладеності. Це дозволяє врахувати ієрархічні властивості AST. Усі ці компоненти об'єднуються конкатенацією, після чого проходять через проєкційний шар, який переводить інформацію в компактний вектор фіксованої розмірності. Такий вектор і є AST Node Embedding.

Отримані ембеддинги виконують подвійну роль у системі. По-перше, вони передаються до GNN, яка аналізує залежності між функціями, файлами або модулями, поширюючи інформацію через структуру графа. Це дозволяє моделі визначити важливі компоненти коду, знайти критичні зв'язки та сформувати узагальнене представлення структури програми.

По-друге, узагальнені графові ембеддинги інтегруються в модель T5 під час генерації коментарів або документації. Завдяки цьому T5 отримує не лише локальний контекст окремого фрагменту коду, але й розуміння його ролі всередині усього проєкту. Такий підхід забезпечує значно більшу точність та інформативність коментарів порівняно з моделями, що працюють виключно з текстом коду.

Таким чином, використання AST Node Embeddings є центральним елементом і основною особливістю всієї розробленої архітектури, бо вони поєднують структурні та семантичні властивості програмного коду у формі, придатній для глибокого навчання, та забезпечують спільну роботу модулів GNN та T5 у рамках запропонованої системи.

4. Опис структури створеної системи

Система, що була розроблена, є комплексним рішенням для аналізу вихідного коду, побудови схем залежностей і автоматичної генерації коментарів до файлів проєкту. Архітектура проєкту включає кілька ключових компонентів, кожен із яких вирішує окремі завдання та інтегрується з іншими частинами системи.

Для реалізації завантаження коду було обрано модуль GitPython, оскільки він надає високорівневий інтерфейс для клонування репозиторіїв, роботи з гілками та вилучення окремих файлів. GitPython обертає функціональність Git у Python-об'єкти, що дозволяє інтегрувати завантаження репозиторіїв у єдину систему без необхідності прямої взаємодії з командним рядком. Такий підхід підвищує переносимість коду та спрощує його інтеграцію з іншими компонентами проєкту, такими як модуль парсингу та графового аналізу.

Для отримання структурної інформації з вихідного коду застосовується ANTLR. За допомогою рекурсивних функцій чи спеціальних Visitor-класів ANTLR дозволяє обійти всі вузли дерева. Це дозволяє отримати структурну інформацію про програму: визначення функцій та класів, аргументи функцій, що використовуються змінні, блоки умовних операторів, цикли. З вузлів AST створюється послідовність токенів, придатна навчання моделі T5. Кожен токен є окремим елементом коду, який може бути зіставлений з коментарями, формуючи пари «код — коментар» для донавчання.

На основі AST можна виявити імпорти модулів та виклики функцій, що використовується для побудови графа залежностей файлів та модулів. Токенізований код, отриманий після розбору AST, використовується як вхідні дані для моделі T5. Перетворення в токени із збереженням структури програми дозволяє моделі «розуміти» контекст функцій та класів, що суттєво підвищує якість автоматичної генерації коментарів.

На основі AST формується граф залежностей, де вузли – це файли чи модулі, а ребра – їх взаємозв'язки через імпорти чи виклики функцій. Для обробки графа використовується Graph Neural Network (GNN) з архітектурою Graph Attention Network (GAT), яка агрегує інформацію від сусідніх вузлів та створює ембеддинги, що відображають контекст кожного вузла. Ці ембеддинги потім передаються T5 і використовуються для візуалізації.

Для візуалізації графа застосовується бібліотека Graph-tool, що дозволяє будувати наочні схеми залежностей. Алгоритми розміщення вузлів (наприклад, `sfdp_layout`) забезпечують зручне представлення графа для аналізу та документування проєкту.

Модель T5-base, донавчена за допомогою методу LoRA, використовується для створення коментарів до вихідного коду. Вхідні дані моделі включають токенизований код, підготовлений на основі AST, та ембединги вузлів графа, що формуються GNN. Це дозволяє моделі враховувати як локальні синтаксичні особливості коду, так і глобальні залежності між файлами, що підвищує точність та інформативність створюваних коментарів. Для токенизації використовується SentencePiece, що оптимально працює з субсловними одиницями і дозволяє обробляти рідкісні ідентифікатори.

Навчання моделі проводиться з використанням PyTorch, оптимізація через AdamW, а функція втрат - Cross-Entropy Loss. Для генерації послідовностей використовується beam search, що дозволяє вибрати найімовірніші токени.

Для підвищення продуктивності та паралельної обробки великих репозиторіїв використовується Celery, що забезпечує асинхронне виконання завдань. Це дозволяє одночасно обробляти кілька репозиторіїв, будувати графи та генерувати коментарі без блокування основних потоків виконання.

Для представлення результатів роботи системи як документації застосовується бібліотека Мако. Вона дозволяє автоматично формувати HTML-документи, що містять візуалізовані графи залежностей із GNN, вихідний код та згенеровані коментарі, а також структуровані звіти щодо кожного модуля. Для взаємодії з користувачем використовується стек Flask та ReactJS.

5. Тестування системи

5.1. Про метрику BERTScore

BERTScore - це сучасна метрика для оцінки якості згенерованого тексту, що використовує попередньо навчені трансформерні моделі, таких як BERT, RoBERTa або їх багатомовні аналоги. На відміну від традиційних метрик на кшталт BLEU або ROUGE, котрі орієнтовані на точний збіг токенів між передбаченим та еталонним текстом, BERTScore враховує семантичну схожість слів та фраз. Це робить метрику більш придатною для завдань генерації природної мови, включаючи коментування коду та рефакторинг документа.

Спочатку і еталонний текст, і згенерований розбиваються на токени, що відповідають обраній моделі трансформера (у даному випадку моделі T5. Для кожного токена обчислюється ембединг за допомогою моделі BERT або її аналога, який можна обрати. Ембединги відображають сенс слова з урахуванням його оточення. Обчислення ембедингу для кожного токена за допомогою трансформера вимагає GPU та великого обсягу пам'яті при великих наборах даних, проте це виправдовується вищою точністю аналізу.

Далі для кожного токена згенерованого тексту знаходиться токен з еталонного тексту, з яким його ембединг має найбільшу косинусну схожість (2).

$$\text{sim}(h_i^g, h_j^r) = \frac{h_i^g \cdot h_j^r}{\|h_i^g\| \cdot \|h_j^r\|} \quad (2)$$

На основі цих пар обчислюються три показники:

- 1) P (Precision) (3) – наскільки згенерований токен схожий на еталон:

$$P = \frac{1}{|x|} \sum_i \max_j \text{sim}(x_i, y_j) \quad (3)$$

- 2) R (Recall) (4) – наскільки еталонний токен «покритий» створеним токеном:

$$R = \frac{1}{|y|} \sum_j \max_i \text{sim}(y_j, x_i) \quad (4)$$

- 3) F1 (5) – гармонічне середнє значення між R та P:

$$F1 = 2 \cdot \frac{P \cdot R}{P + R} \quad (5)$$

де x_i – токени передбачення, y_j – токени еталону, $\text{sim}(x_i, y_j)$ – косинусна схожість між векторами.

Під час аналізу враховуються смислові збіги, а не лише ідентичні слова. Наприклад, "create user" та "add a new user" будуть оцінені як близькі за змістом, хоча структура коментаря та окремі слова відрізняються. Більше того, контекстні ембединги дозволяють відрізнити омоніми та правильно враховувати значення слова у конкретному місці. Проте тестування коротких фрагментів можуть давати нестабільні результати і це слід враховувати. Значення F1 показує

семантичний збіг, але не завжди зрозуміло, які конкретні помилки припустилася моделі генерації тексту.

Також передбачена можливість за потреби використовувати різні трансформери для обчислення ембеддингів, у тому числі багатомовні, що корисно для проектів із кодом та коментарями мовою, відмінною від англійської.

Загалом якість оцінювання залежить від обраної моделі BERT – слабка чи невідповідна модель може знизити точність метрики. Проте дослідження показують, що BERTScore краще корелює з оцінкою якості тексту, виставленої людьми, порівняно з BLEU і ROUGE, що може бути корисним, якщо тестування із залученням людей не передбачене або не є доцільним. У цілому можна стверджувати, що метрика ідеально підходить для автоматизованого тестування якості коментарів та подальшого аналізу результатів експериментів.

5.2. Отримані результати

Для оцінки результату донавчання власної моделі результати було порівняно з T5-base та mT5-base – базовим та багатомовним варіантами моделі T5, результати наведені у таблицях 1 та 2.

Таблиця 1. Порівняння результатів різних моделей

Table 1. Comparison of results of different models

Модель	Precision	Recall	F1
T5-base (базова)	0.851	0.857	0.854
mT5-base (багатомовна)	0.863	0.868	0.865
T5 (донавчена на кодї з коментарями)	0.894	0.901	0.898

Таблиця 2. Порівняння еталонних та згенерованих коментарів

Table 2. Comparison of reference and generated comments

№	Фрагмент коду	Еталонний коментар	T5-base	mT5-base	T5 (донавчена)	BERTScore F1
1	<code>def add(a, b): return a + b</code>	Returns the sum of two numbers.	Adds numbers together.	Returns sum of values.	Computes and returns the sum of two input numbers.	0.93
2	<code>for i in range(len(items)): process(items[i])</code>	Iterates through the list and processes each element.	Loop over list.	Iterates items in list.	Loops through all items and processes each element.	0.89
3	<code>if not os.path.exists(path): os.mkdir(path)</code>	Creates directory if it does not exist.	Check and create folder.	Make directory if missing.	Checks if the directory exists, and creates it if necessary.	0.91
4	<code>user_input = input("Enter your name: ")</code>	Reads user input from console.	Gets input.	Takes user text.	Reads a name entered by the user from the console.	0.88
5	<code>try: open(file) except FileNotFoundError: print("Error")</code>	Handles missing file errors gracefully.	File error handling.	Catch missing file.	Tries to open a file and prints an error if it is not found.	0.90

Результати демонструють, що донавчена модель T5, інтегрована з лексичним аналізом та структурними представленнями коду через AST, стабільно перевищує базову T5 та mT5-small у точності генерації коментарів. Зокрема, при середній довжині функції до 30 рядків спостерігалось підвищення F1 приблизно на 4–5% порівняно з базовою моделлю. Це свідчить про ефективність адаптації моделі до доменної задачі та важливість включення структурної інформації.

Загалом базова англійська модель демонструє задовільні результати на простих текстах, але погано узгоджується з коментарями, що містять терміни програмування. Модель mT5-base завдяки багатомовному переднавчанню модель краще працює із текстами, що містять технічну лексику або синтаксис коду. Проте зберігається певна втрата точності на доменних фразах. Доновчання власної моделі на корпусі з коментарями до коду значно підвищує семантичну узгодженість вихідного тексту з оригінальним коментарем. Модель краще розуміє структурні зв'язки між кодом і природною мовою.

Аналіз залежності точності від складності програмного коду показав, що при збільшенні кількості умовних операторів, вкладених циклів та взаємозв'язків між модулями точність генерації дещо знижується. Наприклад, для функцій із складністю понад 10 умовних або циклічних конструкцій F1 зменшувалась порівняно з більш простими функціями. Однак поєднання AST-представлень та графової моделі GNN дозволяє частково компенсувати цю тенденцію, оскільки система зберігає інформацію про глобальні залежності між компонентами коду.

Досліджено також вплив довжини AST на якість генерації. Було встановлено, що для дуже коротких AST (до 15 вузлів) модель часто пропускає деталі коду, генеруючи лише загальні коментарі. Для середньої довжини AST (15–50 вузлів) точність коментарів є найвищою, оскільки модель здатна повністю захопити локальні та глобальні структури коду. При надзвичайно довгих AST (понад 100 вузлів) спостерігається певне зниження точності, що пов'язано із складністю відстеження всіх зв'язків та обмеженням контексту у трансформерній моделі. Для зменшення цього ефекту у системі застосовуються додаткові механізми обрізання та пріоритетного аналізу ключових вузлів AST.

Висновки

Дане дослідження є актуальним і має суттєву наукову та практичну цінність. Основна особливість роботи полягає у створенні інтегрованої системи генерування документації для програмного коду, яка поєднує донавчену трансформерну модель T5 та графову модель GNN, з використанням абстрактних синтаксичних дерев (AST) у лінійному представленні для аналізу структури коду. Запропонований підхід забезпечує комплексне урахування локальних та глобальних контекстів програмних фрагментів, дозволяючи системі ефективно визначати призначення коду, взаємозв'язки між його компонентами та формувати змістовні коментарі й документацію.

Результати дослідження підтвердили ефективність адаптації трансформерної моделі до конкретної доменної задачі. Доновчання T5 на спеціалізованому корпусі з прокоментованим кодом у поєднанні з лексичним аналізом підвищило якість генерації приблизно на 4% за метрикою F1 порівняно з базовою моделлю.

Це демонструє, що цільове донавчання великих трансформерних моделей може суттєво покращити результати автоматичного документування програмного коду, особливо для великих Python-проектів. Крім того, використання GNN для моделювання залежностей між файлами та модулями дозволяє візуалізувати структуру проекту, забезпечуючи додаткову інформацію для розуміння коду.

Практичне значення результатів полягає у створенні реального інструмента для автоматичного генерування документації, що дозволяє розробникам значно скоротити час на створення пояснень до коду та підтримку великих програмних систем. Використання комбінації «seq2seq» трансформерів, методу донавчання LoRA, токенизації через SentencePiece, аналізу AST і графового моделювання через GNN створює інтегрований підхід, який забезпечує високу точність та гнучкість системи. Така комплексна архітектура дозволяє моделі швидко адаптуватися до нових проектів і специфічних задач.

Наукове значення роботи полягає у тому, що дослідження демонструє можливість об'єднання різних методів машинного навчання для вирішення проблем автоматизації документування коду,

відкриваючи перспективи для подальшого розвитку інтелектуальних систем аналізу програмних проєктів. Досягнуті результати можуть стати основою для розробки нових методів генерації документації, включно з багатомовними системами, інтеграцією з IDE, а також автоматизованим контролем якості програмного забезпечення.

Перспективи подальших досліджень включають удосконалення графових моделей для більш точного відображення залежностей між модулями, оптимізацію архітектури трансформера під специфічні мови програмування, інтеграцію семантичного аналізу коду та підвищення масштабованості системи для великих корпоративних проєктів. Крім того, можлива реалізація адаптивних систем, які самостійно оновлюють документацію під час внесення змін у код, що значно підвищить продуктивність розробників і зменшить ризики помилок у великих програмних системах.

Таким чином, результати роботи демонструють ефективність поєднання трансформерних та графових моделей, а також забезпечують комплексний підхід до автоматизації створення документації. Запропонована система має потенціал для широкого практичного застосування та подальшого розвитку у сфері автоматичного аналізу коду й інтелектуальної підтримки програмної розробки.

REFERENCES

1. Van Heesch D. *Doxygen: Source Code Documentation Generator* [Electronic resource]. – Available at: <https://www.doxygen.nl/>
2. GitHub. *GitHub Copilot Documentation* [Electronic resource]. – Available at: <https://docs.github.com/copilot>
3. Swimm. *Auto-Docs: AI-Powered Code Documentation Generator* [Electronic resource]. – Available at: <https://swimm.io/>
4. Shiina H., Onishi S., Takahashi A., Kobayashi N. Automatic comment generation for source code using external information by neural networks for computational thinking // *International Journal of Smart Computing and Artificial Intelligence*. — 2021. — Vol. 5, No. 2. — P. 15–28.
5. Wei B., Li G., Xia X., Fu Q., Jin Z. Retrieve and Refine: Exemplar-based Neural Comment Generation // *Proceedings of the 27th International Conference on Software Analysis, Evolution and Reengineering (SANER)*. — 2020. — P. 1–12.
6. Krasanakis A., Papadopoulos S., Kompatsiaris I. JGNN: Graph Neural Networks on Native Java // *Proceedings of the 20th International Conference on Mining Software Repositories (MSR)*. — 2023. — P. 1–12.
7. Vaswani A., Shazeer N., Parmar N. et al. Attention Is All You Need // *Advances in Neural Information Processing Systems*. – 2017. – № 30. – Available at: <https://arxiv.org/abs/1706.03762>
8. Raffel C., Shazeer N., Roberts A. et al. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer (T5) [Electronic resource]. – 2020. – Available at: <https://arxiv.org/abs/1910.10683>
9. Hu E. J., Shen Y., Wallis P. et al. LoRA: Low-Rank Adaptation of Large Language Models [Electronic resource]. – 2021. – Available at: <https://arxiv.org/abs/2106.09685>
10. Zhang T., Kishore V., Wu F. et al. BERTScore: Evaluating Text Generation with BERT [Electronic resource]. – 2020. – Available at: <https://arxiv.org/abs/1904.09675>

**Albina
Kostiuchenko**

*Master's student of the Department of System Programming and Specialized Computer Systems, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 03056, Ukraine, Kyiv, Polytechnichna St., 14-a
e-mail: albina.kostyuchenko03@gmail.com
<https://orcid.org/0009-0004-7382-7209>*

**Andrii
Petrashenko**

*Ph.D., Associate Professor of the Department of System Programming and Specialized Computer Systems, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 03056, Ukraine, Kyiv, Polytechnichna St., 14-a
e-mail: petrashenko@gmail.com;
<https://orcid.org/0000-0003-0239-1706>*

Method for generating source code description using an artificial intelligence model

Relevance. The topic is relevant, since currently there are many large projects that are being developed over a long period of time and require support and understanding of the code without explanations. The rapid development of technologies and the need to constantly develop new features and support existing ones require constant updating of documentation. Writing good documentation is a valuable skill that requires experience, concentration and understanding of the project structure. As a result, a large number of developers consider the process of writing documentation difficult and think that the time spent on it could be used more productively. That is why there is a demand for services that help automate this process.

Goal. The purpose of this work is to increase the efficiency of automated generation of software documentation. As part of this task, the necessary theoretical material was worked out, existing solutions to this problem were studied, and our own new method of generating a description of the program code was developed and implemented, which more accurately determined the purpose of code fragments, clearly understood the structure and dependencies between its components.

Research methods. The study is based on literature analysis, statistical methods, as well as machine learning and data mining methods. In particular, the methods of syntactic code analysis and construction of an abstract syntax tree (AST), the method of forming a training corpus, methods of training and retraining of transformer and graph models were used. To assess the advantages of the retrained model, the method of comparative modeling and automated text quality assessment (in this case, BERTScore) was used.

The results. Retraining the T5 model on a specialized dataset with commented code in combination with lexical analysis allowed to increase the quality of generation by approximately 4% in terms of the F1 metric compared to the base model. This indicates that adapting the model to a specific domain task is effective and can significantly improve the result.

Conclusions. Based on the collected data, an own approach was proposed to improve the quality of code description generation using the retrained T5 model and the created GNN model with further implementation, which is the result of the research. The proposed system combines the best practices of syntactic analysis, graph modeling, and transformer generation, providing a practically applicable solution for automatic documentation creation. It can be argued that the combination of "seq2seq" models, tokenization and adaptation methods of large transformers, as well as code analysis via GNN and structural AST representations provides a comprehensive approach to automating work with code, allowing you to combine local and global contexts, quickly adapt the model to specific tasks, and effectively generate meaningful comments and documentation. Such an integrated approach has the potential for further development of artificial intelligence systems in the field of automatic code analysis, increasing developer productivity, and ensuring software quality. The research results can be applied in practice for fast and effective creation of documentation for developed software and large projects in the Python language.

Keywords: *machine learning, T5, GNN, code description generator, model training, natural language processing, documentation, AST.*

UDC 517.9:519.6

Koshkin Andrii

assistant of the Department of Applied Mathematics
Kharkiv National University of Radio Electronics, Nauky Ave. 14,
Kharkiv, Ukraine, 61166
e-mail: andrii.koshkin@nure.ua;
<https://orcid.org/0009-0005-0970-0403>

Strelnikova Olena

Professor, Doctor of Science (Engineering)
Kharkiv National University of Radio Electronics, Nauky Ave. 14,
Kharkiv, Ukraine, 61166
leading researcher
Anatolii Pidhornyi Institute of Power Machines and Systems of NAS of
Ukraine, Komunalnykiv street 2/10, Kharkiv, Ukraine, 61046
e-mail: olena.strelnikova@nure.ua;
<https://orcid.org/0000-0003-0707-7214>

Bending analysis of multiply-connected anisotropic plates with elastic inclusions

Relevance. Determining the stress-strain state of thin anisotropic plates with foreign elastic inclusions under transverse bending is an important engineering problem. However, the general case of a plate with multiple, arbitrarily arranged inclusions has lacked an effective numerical or analytical solution due to significant mathematical and computational difficulties.

Objective. The purpose of this work is to develop a new approximate method for determining the stress state of a thin anisotropic plate containing a group of arbitrarily located elliptical or linear elastic inclusions.

Methods. The method is based on the application of S. G. Lekhnitskii's complex potentials. The problem is reduced to determining functions of generalized complex variables for the plate-matrix and the inclusions. These potentials are represented by corresponding Laurent series and Faber polynomials. The generalized least squares method (GLSM) is used to satisfy the contact boundary conditions on the inclusion contours. This reduces the problem to an overdetermined system of linear algebraic equations, which is solved using singular value decomposition (SVD).

Results. The developed method was validated by comparison with the known exact analytical solution for a plate with a single elliptical inclusion, showing perfect agreement. Numerical studies were conducted to analyze the influence of the relative stiffness of the inclusions, the distances between them, and their geometric characteristics on the bending moment values. It was established that the interaction between inclusions is significant and leads to a substantial increase in moments at small distances. Isotropic plates are considered as a special case of anisotropic ones.

Conclusions. It was established for the first time that for linear elastic inclusions, moment singularities, described by moment intensity factors (MIFs), occur only in cases of sufficiently stiff or sufficiently flexible inclusions.

Keywords: thin plate, inclusions, cracks, complex potentials, boundary value problem, mathematical modeling, numerical methods, moment intensity factors.

How to quote: A. Koshkin, and O. Strelnikova, "Bending analysis of multiply-connected anisotropic plates with elastic inclusions", *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 68, pp. 43-53, 2025. <https://doi.org/10.26565/2304-6201-2025-68-04>

Як цитувати: Koshkin A., and Strelnikova O. Bending analysis of multiply-connected anisotropic plates with elastic inclusions. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління*. 2025. вип. 68. С.43-53. <https://doi.org/10.26565/2304-6201-2025-68-04>

1. Introduction

Despite the significant practical importance of determining the stress state of thin plates under transverse bending, especially in case of presence of foreign inclusions, many engineering problems remain unsolved. It is connected with major mathematical and computational difficulties that have been historically challenging to overcome.

Significant results in developing solution methods for the applied theory of bending of anisotropic plate were obtained by S. G. Lekhnitskii [1] as early as the mid-1930s. Complex potentials were introduced, and the simplest problems for simply-connected domains were solved. Somewhat later, methods for solving problems for multiply-connected plates were developed [2]. However, inaccuracies in the general representations of complex potentials in [2], the use of the series method to satisfy boundary conditions in such complex problems, and the limited computational technology of that time prevented the solution of many relevant problems in plate bending theory. To satisfy the boundary conditions in such complex problems, the discrete least squares method was initially applied, followed by the generalized least squares method (GLSM). This evolution in solution methods allowed to analyse any multiply-connected domains with arbitrarily shaped contours. The GLSM, in particular, proved to be the simplest for computer implementation.

The presence of elastic inclusions in plates introduces additional difficulties in solving these problems. For a plate with elliptical (circular) elastic inclusions under a generalized plane stress state, a large number of problems have been solved [2–4]; plates with linear inclusions have been considered as a special case [5]. In the case of transverse bending of plate, such problems have only been solved for a plate with a single elliptical (circular) inclusion [1, 6] or a single linear inclusion [7]. However, the general case of the bending of a multiply-connected plate with several elastic inclusions, including linear ones, has not been considered.

This article, using the GLSM, presents for the first time a solution to the problem of the bending of an anisotropic plate containing multiple elliptical or linear elastic inclusions. The results of numerous calculations are described, which identify the influence of the elastic inclusions' stiffness, the distances between them, and the geometric characteristics of the inclusions themselves on the bending moment values arising in the plate. Isotropic plates are considered as a special case of anisotropic ones. To simplify the utilized relations, the plate is assumed to be infinite, and the surfaces of the plate and inclusions are unloaded.

2. Problem Formulation and Solution Method

Let us consider an infinite anisotropic plate-matrix, occupying a multiply-connected domain S , bounded by the contours L_l ($l = \overline{1, L}$) of elliptical holes. These holes have semi-axes a_l , b_l , centers at points O_l (with coordinates x_{0l} , y_{0l}), and orientation angles φ_l formed by the x -axis with the semi-axes a_l (Fig. 2.1). Elastic inclusions, occupying domains $S^{(l)}$ and made of different materials, are inserted into these holes without pre-stress. These inclusions are in perfect mechanical contact with the plate. At infinity, the plate is subjected to moments M_x^∞ , M_y^∞ and H_{xy}^∞ .

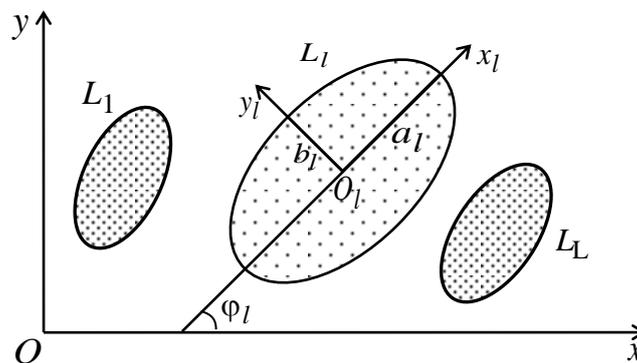


Fig. 2.1 Infinite anisotropic plate with inclusions

Рис. 2.1 Нескінченна анізотропна плита з включеннями

Let's use the complex potentials from the applied theory of anisotropic plate bending. It leads to finding the complex potentials $W_k'(z_k)$ for the plate-matrix and $W_k^{(l)'}(z_k^{(l)})$ for the inclusions, based on the corresponding boundary conditions.

The complex potentials $W'_k(z_k)$ for the plate-matrix are functions of the generalized complex variables

$$z_k = x + \mu_k y, \quad (2.1)$$

where μ_k are the roots of the characteristic equation

$$D_{22}\mu^4 + 4D_{26}\mu^3 + 2(D_{12} + 2D_{66})\mu^2 + 4D_{16}\mu + D_{11} = 0, \quad (2.2)$$

$D_{ij} = B_{ij}D_0$ are the flexural rigidities of the plate material, where

$$\begin{aligned} B_{11} &= (a_{22}a_{66} - a_{26}^2)/\Delta, & B_{12} &= (a_{16}a_{26} - a_{12}a_{66})/\Delta, & B_{16} &= (a_{12}a_{26} - a_{16}a_{22})/\Delta \\ B_{22} &= (a_{11}a_{66} - a_{16}^2)/\Delta, & B_{26} &= (a_{12}a_{16} - a_{26}a_{11})/\Delta, & B_{66} &= (a_{11}a_{22} - a_{12}^2)/\Delta, \\ \Delta &= \begin{vmatrix} a_{11} & a_{12} & a_{16} \\ a_{12} & a_{22} & a_{26} \\ a_{16} & a_{26} & a_{66} \end{vmatrix}; \end{aligned}$$

a_{ij} are the compliance coefficients of the material; $D_0 = 2h^3/3$; h is the half-thickness of the plate. These functions are defined in the domains S_k , obtained from the original domain S by the affine transformations (2.1), and in our case they can be expressed in next form:

$$W'_k(z_k) = \Gamma_k z_k + \sum_{l=1}^L \sum_{p=1}^{\infty} \frac{a_{klp}}{\zeta_{kl}^p}, \quad (2.3)$$

in which Γ_k are constants determined from the system of equations

$$\begin{aligned} 2\operatorname{Re} \sum_{k=1}^2 \Gamma_k &= C_{11}M_x^\infty + C_{21}M_y^\infty + C_{31}H_{xy}^\infty, \\ 2\operatorname{Re} \sum_{k=1}^2 \mu_k \Gamma_k &= C_{12}M_x^\infty + C_{22}M_y^\infty + C_{32}H_{xy}^\infty, \\ 2\operatorname{Re} \sum_{k=1}^2 \mu_k^2 \Gamma_k &= C_{13}M_x^\infty + C_{23}M_y^\infty + C_{33}H_{xy}^\infty, & 2\operatorname{Re} \sum_{k=1}^2 \frac{1}{\mu_k} \Gamma_k &= 0; \\ C_{11} &= (2D_{22}D_{66} - 2D_{26}^2)/\Delta_1, \\ C_{21} &= (2D_{16}D_{26} - 2D_{12}D_{66})/\Delta_1, \\ C_{31} &= (2D_{12}D_{26} - 2D_{12}D_{22})/\Delta_1, \\ C_{12} &= (D_{12}D_{26} - D_{16}D_{22})/\Delta_1, \\ C_{22} &= (D_{12}D_{16} - D_{11}D_{26})/\Delta_1, \\ C_{32} &= (D_{11}D_{22} - D_{12}^2)/\Delta_1, \\ C_{13} &= (2D_{16}D_{26} - 2D_{12}D_{66})/\Delta_1, \\ C_{23} &= (2D_{11}D_{66} - 2D_{16}^2)/\Delta_1, \\ C_{33} &= (2D_{12}D_{16} - 2D_{11}D_{26})/\Delta_1, \\ \Delta_1 &= \begin{vmatrix} D_{11} & 2D_{16} & D_{12} \\ D_{12} & 2D_{26} & D_{22} \\ D_{16} & 2D_{66} & D_{26} \end{vmatrix}; \end{aligned} \quad (2.4)$$

ζ_{kl} are variables obtained from the conformal mappings of the exterior of the unit circle $|\zeta_{kl}| \geq 1$ onto the exterior of the ellipses L_{kl}

$$z_k = z_{kl} + R_{kl} \left(\zeta_{kl} + \frac{m_{kl}}{\zeta_{kl}} \right); \quad (2.5)$$

$$\begin{aligned} z_{kl} &= x_{0l} + \mu_k y_{0l}, \\ R_{kl} &= \left[a_l (\cos \varphi_l + \mu_k \sin \varphi_l) + ib_l (\sin \varphi_l - \mu_k \cos \varphi_l) \right] / 2, \\ m_{kl} &= \left[a_l (\cos \varphi_l + \mu_k \sin \varphi_l) - ib_l (\sin \varphi_l - \mu_k \cos \varphi_l) \right] / 2R_{kl}; \end{aligned} \quad (2.6)$$

a_{klp} are unknown constants.

Once the complex potentials are determined, the plate deflection, bending moments and transverse shear forces are calculated using the formulas [1, 2]:

$$\begin{aligned} w &= 2 \operatorname{Re} \sum_{k=1}^2 W_k(z_k), \\ (M_x, M_y, H_{xy}, N_x, N_y) &= -2 \operatorname{Re} \sum_{k=1}^2 (p_k, q_k, r_k, \mu_k s_k, -s_k) W_k''(z_k), \end{aligned} \quad (2.7)$$

in which

$$\begin{aligned} p_k &= D_{11} + 2D_{16}\mu_k + D_{12}\mu_k^2, \\ q_k &= D_{12} + 2D_{26}\mu_k + D_{22}\mu_k^2, \\ r_k &= D_{16} + 2D_{66}\mu_k + D_{26}\mu_k^2, \\ s_k &= -D_{16} - (D_{12} + 2D_{66})\mu_k - 3D_{26}\mu_k^2 - D_{22}\mu_k^3. \end{aligned} \quad (2.8)$$

For the bending moments on elements with a normal n we have:

$$M_n = M_x \cos^2 nx + M_y \cos^2 ny + 2H_{xy} \sin nx \cos ny. \quad (2.9)$$

The complex potentials $W_k^{(l)}(z_k)$ for the inclusions ($l = \overline{1, L}$) are functions of the generalized complex variables

$$z_k^{(l)} = x + \mu_k^{(l)} y, \quad (2.10)$$

where $\mu_k^{(l)}$ are the roots of characteristic equations of type (2.2), in which the coefficients D_{ij} are replaced by the constants $D_{ij}^{(l)}$ for the inclusions. These functions are defined in the domains $S_k^{(l)}$ obtained from the domains $S^{(l)}$ by the affine transformations (2.10). They are holomorphic in the domains $S_k^{(l)}$ and can be expanded in series of Faber polynomials, which, after transformations, are represented as power series:

$$W_k^{(l)}(z_k^{(l)}) = \sum_{p=0}^{\infty} a_{kp}^{(l)} \left(\frac{z_k^{(l)} - z_{kl}^{(l)}}{R_{kl}^{(l)}} \right)^p. \quad (2.11)$$

Here $R_{kl}^{(l)}$ and $z_{kl}^{(l)}$ are constants calculated by formulas (2.6) for R_{kl} and z_{kl} respectively, substituting μ_k for $\mu_k^{(l)}$; $a_{kp}^{(l)}$ are unknown constants.

The unknown constants a_{klp} and $a_{kp}^{(l)}$ will be determined from the boundary conditions on the contours of the plate and the inclusions:

$$2 \operatorname{Re} \sum_{k=1}^2 \left(g_{kli} W_k'(z_k) - g_{kli}^{(l)} W_k^{(l)'}(z_k) \right) = f_{li} \quad (i = \overline{1, 4}), \quad (2.12)$$

in which

$$\begin{aligned} g_{kl1} &= 1, \quad g_{kl1}^{(l)} = 1, \quad g_{kl2} = \mu_k, \quad g_{kl2}^{(l)} = \mu_k^{(l)}, \\ g_{kl3} &= p_k / \mu_k, \quad g_{kl3}^{(l)} = p_k^{(l)} / \mu_k^{(l)}, \quad g_{kl4} = q_k, \quad g_{kl4}^{(l)} = q_k^{(l)}; \\ f_{11} &= f_{12} = 0, \quad f_{13} = -c_{1x} + c_{1l}, \quad f_{14} = -c_{1y} + c_{2l}, \end{aligned}$$

where c_l are real constants and c_{1l}, c_{2l} are complex constants.

When using the generalized least squares method GLSM, we will satisfy the boundary conditions (2.12) in a differential form in order to eliminate the arbitrary complex constants within them. To do this, we select a set of collocation points $t_{lm} = x_{lm} + iy_{lm}$ ($m = \overline{1, M_l}$) on the plate-inclusion interface. Then we substitute the functions (2.3) and (2.11) into the boundary conditions (2.12) after differentiating them with respect to the contour arc. This yields a system of linear equations:

$$2\text{Re} \sum_{k=1}^2 \delta_k \left[\sum_{l=1}^L \sum_{p=1}^{\infty} g_{kli} \phi'_{klp}(t_{km}) a_{klp} - \sum_{p=1}^{\infty} g_{kp}^{(l)} \phi_{kp}^{(l)}(t_{km}) a_{kp}^{(l)} \right] = \frac{df_{li}(t_{km})}{ds} - 2\text{Re} \sum_{k=1}^2 \delta_k g_{kli} \Gamma_k \quad (l = \overline{1, L}; m = \overline{1, M_l}; i = \overline{1, 4}). \tag{2.13}$$

Here,

$$\phi'_{klp} = -\frac{p}{\zeta_{kl}^{p-1} (\zeta_{kl}^2 - m_{kl}) R_{kl}}; \quad \phi_{kp}^{(l)} = \frac{p(z_k^{(l)} - z_{kl})^{p-1}}{(R_k^{(l)})^p};$$

$$\delta_k = dt_k / ds; \quad t_{km} = x_m + \mu_k y_m, \quad t_{km}^{(l)} = x_m + \mu_k^{(l)} y_m.$$

The system (2.13) is supplemented by the single-valuedness conditions for the deflection function $w(x, y)$ for each hole in the plate-matrix

$$2\text{Re} \sum_{k=1}^2 ia_{jkli} R_{kl} = 0 \quad (l = \overline{1, L}). \tag{2.14}$$

After finding the pseudo-solutions of the augmented system (2.13) using singular value decomposition (SVD) [8], the constants a_{klp} , $a_{kp}^{(l)}$, and consequently, the functions (2.3) and (2.11), become known. The bending moments and shear forces (2.7) can then be determined from them. If an inclusion $S^{(l)}$ is reduced into a linear elastic line (and the corresponding hole into a slit), the moment intensity factors (MIFs) k_1^{\pm} (for moments $M_y^{(l)}$ in the local coordinate system centered at point O_l (see Fig. 2.1)) and k_2^{\pm} (for moments $H_{xy}^{(l)}$ in the same coordinate system) can also be calculated. The superscripts – and + refer to the left and right tips of the inclusion respectively. By analogy with the derivation of stress intensity factors (SIFs) for slit tips in plates [9], the formulas for them are:

$$k_1 = 2\text{Re} \sum_{k=1}^2 \left[p_k \sin^2 \varphi_l + q_k \cos^2 \varphi_l - r_k \sin 2\varphi_l \right] M_k,$$

$$k_2 = 2\text{Re} \sum_{k=1}^2 \left[\frac{1}{2} (q_k - p_k) \sin 2\varphi_l + r_k \cos 2\varphi_l \right] M_k, \tag{2.15}$$

It should be noted that in the case of a plate with a single elliptical inclusion, an exact analytical solution can also be obtained using the series method, in the form of:

$$W'_k(z_k) = \Gamma_k z_k + \frac{a_{k11}}{\zeta_{k1}}, \quad W_k^{(1)}(z_k) = a_{k1}^{(1)} \frac{z_k^{(1)}}{R_k^{(1)}},$$

$$(M_x, M_y, H_{xy}) = -2\text{Re} \sum_{k=1}^2 (p_k, q_k, r_k) \left(\Gamma_k - \frac{a_{k11}}{R_{k1} (\zeta_{k1}^2 - m_{k1})} \right),$$

$$(k_1^{\pm}, k_2^{\pm}) = \frac{1}{\sqrt{a_1}} 2\text{Re} \sum_{k=1}^2 (q_k, r_k) a_{k11}, \tag{2.16}$$

in which a_{k11} and $a_{k1}^{(1)}$ are constants determined from a system of linear algebraic equations of the 4th order.

3. Numerical studies

Numerical studies were conducted for a plate made of KAST-V isotropic material (Material M1) and angle-ply fiberglass (Material M2). The compliance coefficients for these materials are presented in Table 3.1. In the numerical calculations, the compliance coefficients for the inclusion material were selected as follows: $a_{ij}^{(l)} = \lambda^{(l)} a_{ij}$, where $\lambda^{(l)}$ is the relative stiffness parameter of the inclusion $S^{(l)}$.

Table 3.1. The compliance coefficients of materials
Таблиця 3.1. Коефіцієнти деформації матеріалів

Material	$a_{11} \cdot 10^{-4}, MPa^{-1}$	$a_{22} \cdot 10^{-4}, MPa^{-1}$	$a_{12} \cdot 10^{-4}, MPa^{-1}$	$a_{66} \cdot 10^{-4}, MPa^{-1}$
M1	72,100	72,100	-8,600	161,500
M2	10000	2,800	-0,770	27,000

During the numerical studies, the number of terms N in the series functions (2.3) and (2.11) and the number of collocation points M_l on the plate-inclusion interface (where conditions (2.13) were satisfied) were increased until the boundary conditions were met with a sufficiently high degree of accuracy. The calculations showed that, depending on the distances between inclusions and the semi-axis ratios of the ellipses, satisfying the boundary conditions required retaining from 7 to 20 terms in the series for each hole and inclusion, and selecting from 50 to 1000 collocation points on each contour. Below, some of the obtained results are described for the case where, at infinity, $M_y^\infty = m_y$ and $M_x^\infty = H_{xy}^\infty = 0$. All results are presented as multiples of m_y / D_0 . For the case of a plate with a single inclusion, the exact solution (2.16) was also used for comparison.

It was established that the moment values are significantly influenced by the semi-axis ratio of the ellipses. Calculations showed that for $b_1/a_1 < 10^{-3}$ the inclusion can be considered linear, and MIFs (Moment Intensity Factors) can be calculated for it. In Table 3.2, the M_s moment values for a plate with a single inclusion are presented. These values are shown for points on the plate near the inclusion and for various semi-axis ratios b_1/a_1 and relative stiffness parameters $\lambda^{(l)}$. For the case of a linear inclusion ($b_1/a_1 = 10^{-4}$), the MIFs are also presented.

Table 3.2. The M_s moment values for a plate with a single inclusion
Таблиця 3.2. Значення моментів M_s для плити з одним включенням

Material	$\lambda^{(l)}$	b_1/a_1						MIF k_1^\pm
		1	0,5	10^{-1}	10^{-2}	10^{-3}	10^{-4}	
M1	0	-0,16	-0,19	-0,42	-3,02	-29,00	-288,69	-0,136
	10^{-2}	-0,12	-0,16	-0,33	-0,97	-1,32	-1,38	-0,001
	0,5	0,72	0,63	0,53	0,50	0,49	0,49	0,000
	2	1,25	1,40	1,77	1,97	1,99	1,99	0,000
	10^2	1,70	2,40	7,62	42,10	86,81	97,28	0,013
	∞	1,72	2,44	8,18	72,78	718,83	7178,87	1,000
M2	0	-0,22	-0,27	-0,69	-5,37	-52,18	-520,13	-0,170
	10^{-2}	-0,19	-0,24	-0,53	-1,51	-1,99	-2,06	-0,001
	0,5	0,65	0,58	0,51	0,49	0,49	0,49	0,000
	2	1,37	1,55	1,86	1,97	1,99	1,99	0,000
	10^2	2,31	3,59	12,71	57,10	91,35	97,22	0,007
	∞	2,35	3,70	14,48	135,77	1348,63	13475,67	1,000

It is evident that as the ratio b_1/a_1 decreases, the absolute values of the moments M_s near the tips of the major semi-axis a_1 increase sharply, and for small b_1/a_1 values, this indicates the singular behavior characterized by MIFs.

Fig. 3.1 shows plots of the MIF (k_1^\pm) variation as a function of the inclusion material stiffness (parameter $\lambda^{(1)}$). It can be seen that for a linear inclusion, when $\lambda^{(1)} < 10^{-3}$, the inclusion can be considered perfectly rigid; when $\lambda^{(1)} > 10^3$, it is considered perfectly flexible (a crack). For $10^{-3} < \lambda^{(1)} < 10^3$ the MIF values are very small and can be neglected. Therefore, it is not meaningful to discuss MIFs for linear elastic inclusions when their material stiffness differs from that of the plate material by less than a factor of 10^3 , although other authors, using different models and methods, have reached different conclusions [7]. It should also be noted that for perfectly flexible linear inclusions (cracks), their faces will come into contact. This must be accounted for when solving such problems, for instance, by applying appropriate additional tensile forces. These issues are not addressed in this study. We note that the exact solution (2.16) was also used for the plate with a single inclusion. The results obtained from the approximate GLSM solution were found to coincide with the results from the exact solution.

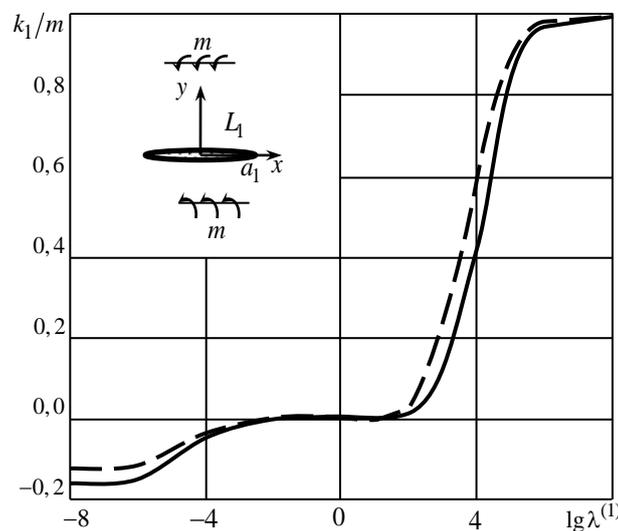


Fig. 3.1 MIF variation depending on inclusion stiffness

Рис. 3.1 Зміна КІМ в залежності від жорсткості включення

Table 3.3 presents the bending moment values M_s for an infinite plate with two identical circular inclusions (Fig. 3.2) of radius a_1 ($b_1 = a_2 = b_2 = a_1$). The values are given for various distances c between the inclusions and various values of the relative stiffness parameter $\lambda^{(1)}$ (where $\lambda^{(2)} = \lambda^{(1)}$). The moments are calculated at the contact points of the plate with the left inclusion, on elements perpendicular to the inclusion contour. Here, θ is the central angle for the left inclusion, measured counter-clockwise from the line of centers.

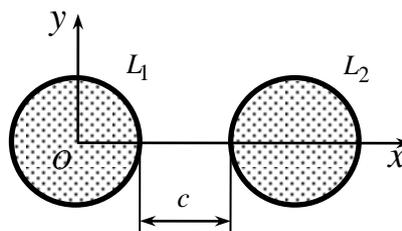


Fig. 3.2 Infinite anisotropic plate with two circular inclusions

Рис. 3.2 Нескінченна анізотропна плита з двома круговими включеннями

The data in Table 3.3 show that as the distance between the inclusions decreases, the M_s moment values at points near the ligament (the area between the inclusions) increase. If the distance between the inclusions is greater than the diameter of one of them ($c/a_1 > 2$), the influence of one inclusion on the stress state around the other is insignificant and can be neglected. As the stiffness of the inclusions decreases (with an increase of $\lambda^{(1)}$), the M_s moment values at the point corresponding to $\theta = 0$ (at the ligament) increase. At the point corresponding to $\theta = \pi/2$, the moments decrease for $\lambda^{(1)} > 1$ (i.e., stiffer inclusions) and increase for $\lambda^{(1)} < 1$ (i.e., more flexible inclusions).

Table 3.3. The M_s moment values for a plate with two inclusions

Таблиця 3.3. Значення моментів M_s для плити з двома включеннями

$\lambda^{(1)}$	c/a_1	θ, rad					
		0	$\pi/2$	π	0	$\pi/2$	π
		M1			M2		
0	∞	-0,16429	0,37740	-0,16429	-0,22231	0,19019	-0,22231
	2	-0,15829	0,31479	-0,08789	-0,24754	0,17038	-0,14703
	1	-0,12078	0,26235	-0,01533	-0,26818	0,15308	-0,04814
	0.5	-0,04908	0,21283	0,03215	-0,28425	0,13658	0,05675
	0.1	0,12031	0,18256	0,00779	-0,43370	0,12183	0,10487
10^{-2}	∞	-0,12351	0,35057	-0,12351	-0,18512	0,17639	-0,18512
	2	-0,12717	0,29375	-0,05347	-0,21340	0,15806	-0,11195
	1	-0,07491	0,24712	0,01489	0,22438	0,09538	0,33411
	0.5	-0,15650	0,18608	0,04670	0,22850	0,05939	0,55252
	0.1	-0,13326	0,17431	0,14177	0,41496	0,11151	0,74618
0,5	∞	0,71970	-0,00208	0,71970	0,65308	-0,00633	0,65308
	1	0,61813	-0,00630	0,72212	0,58172	-0,00825	0,66258
	0.1	0,53298	-0,01874	0,74171	0,39733	-0,01669	0,72451
2	∞	1,24827	0,05753	1,24827	1,36933	0,04577	1,36933
	1	1,39080	0,05634	1,26801	1,47429	0,04405	1,38645
	0.1	1,70994	0,06741	1,26061	1,89385	0,05277	1,35439
10^2	∞	1,70384	0,27407	1,70384	2,31079	0,25327	2,31079
	2	1,92563	0,26062	1,77606	2,48063	0,24013	2,38868
	1	2,24528	0,26094	1,82188	2,76070	0,23748	2,45147
	0.5	2,80183	0,26642	1,86998	3,38583	0,23875	2,52330
	0.1	4,23279	0,29051	1,93290	2,54783	0,18390	1,99577
∞	∞	1,71784	0,28251	1,71784	2,34765	0,26278	2,34765
	2	1,94561	0,26859	1,79201	2,52289	0,24917	2,42875
	1	2,27663	0,26857	1,83915	2,81262	0,24625	2,49507
	0.5	2,86793	0,27381	1,88746	3,45585	0,24880	2,57465
	0.1	5,62046	0,28538	1,97168	7,32184	0,26064	2,74416

Numerical studies were also conducted for the case of two linear inclusions. Fig. 3.3 shows the variation of the MIF ratio k_1^+/k_{10}^+ for the isotropic material M1, where k_1^+ represents the MIF at the inner tip (right tip of the left inclusion) and k_{10}^+ is the baseline MIF for an isolated single inclusion. For material M2, the corresponding graph differs only slightly from that shown in Fig. 3.3, and therefore it is not depicted. It is evident that as the distance between the linear inclusions decreases, the MIFs for the inclusions' inner tips increase significantly.

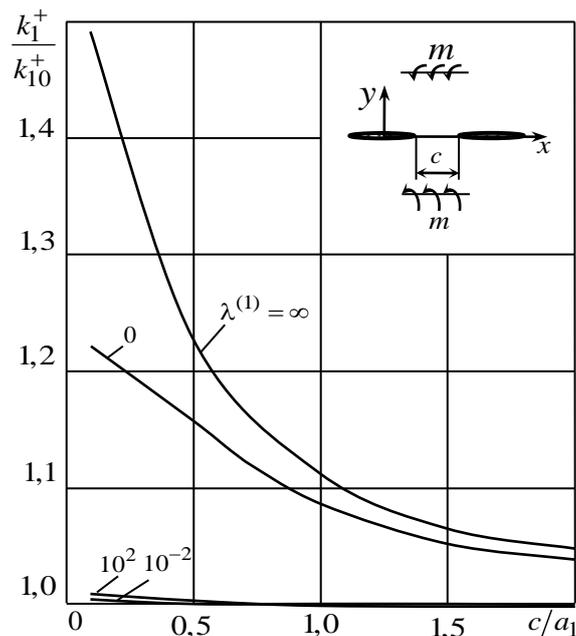


Fig. 3.3 Variation of the MIF ratio for case of two linear inclusions
 Рис. 3.3 Зміна відношення КІМ для випадку двох лінійних включень

4. Conclusions

In this paper, a new effective approximate method for analyzing the stress state of thin anisotropic plates with multiple elastic inclusions under bending has been developed. The approach is based on the application of complex potentials, their representation by Laurent series and Faber polynomials, and the implementation of the Generalized Least Squares Method to satisfy the contact boundary conditions. The problem is reduced to solving an overdetermined system of linear algebraic equations using Singular Value Decomposition (SVD).

The high accuracy and reliability of the proposed method were validated by comparing its results with the known exact analytical solution for the case of a single elliptical inclusion, demonstrating perfect agreement.

The primary scientific contributions of this work are:

1. A robust computational framework capable of handling multiple, arbitrarily oriented elliptical or linear inclusions, a problem not previously solved in the general case.

2. A key finding regarding linear inclusions: it was established that moment singularities, described by MIFs, arise only for sufficiently stiff ($\lambda^{(1)} < 10^{-3}$) or sufficiently flexible ($\lambda^{(1)} > 10^3$) inclusions.

3. The significant effect of inclusion interaction was quantified. For both circular and linear inclusions, it was shown that as the distance between them decreases, the stress (moment) concentration in the ligament region increases substantially.

The developed method demonstrates significant potential for further development. Future research could focus on several promising directions:

- Analyzing periodic or finite arrays (rows and bands) of inclusions to effectively model the mechanical behavior of modern composite structures.
- Extending the method to analyze inclusions of arbitrary (non-elliptical) shapes, which are common in engineering practice, by employing advanced conformal mapping techniques and polynomial approximations.

REFERENCES

1. S.G. Lekhnitskii, S.W. Tsai and T. Cheron, *Anisotropic Plates*. New York: Gordon and Breach, 1968, 534 p. <https://books.google.com.ua/books?id=Ukl9AAAAIAAJ>
2. G.N. Savin, *Stress distribution around holes*. Washington, D.C.: NASA TT, 1970, 997 p. <https://books.google.com.ua/books?id=eC9e0QEACAAJ>

3. C. Hwu and J. Yen Wen, "On the Anisotropic Elastic Inclusions in Plane Elastostatics". *J. Appl Mech*, vol. 60 (3), pp. 626–632, 1993. <https://doi.org/10.1115/1.2900850>
4. C. Hwu, *Anisotropic Elastic Plates*. New York: Springer, 2010, 673 p. <https://doi.org/10.1007/978-1-4419-5915-7>
5. J. Lee, "Elastic analysis of unbounded solids using volume integral equation method". *J Mech Sci Technol*, vol. 22, pp. 450–459, 2008. <https://doi.org/10.1007/s12206-007-1215-2>
6. M.C. Hsieh and C. Hwu, "Anisotropic elastic plates with holes / cracks / inclusions subjected to out-of-plane bending moments". *Int. J. Solids and Struct*, vol. 39, no. 19, pp. 4905–4925, 2002. [https://doi.org/10.1016/S0020-7683\(02\)00335-9](https://doi.org/10.1016/S0020-7683(02)00335-9)
7. O.V. Maksymovych, T.Y. Solyar and Y. Kempa, "Investigation of Bending of Anisotropic Plates with Inclusions with the Help of Singular Integral Equations". *J Math Sci*, vol. 254, pp. 129–141, 2021. <https://doi.org/10.1007/s10958-021-05293-7>
8. Z. Drmač and K. Veselić, "New fast and accurate Jacobi SVD algorithm. I". *SIAM J. Matrix Anal. Appl*, vol. 29, no. 4, pp. 1322–1342, 2008. <https://doi.org/10.1137/050639193>
9. G.C. Sih, P.C. Paris and G.R. Irwin, "On cracks in rectilinearly anisotropic bodies". *Int J Fract*, vol. 1, pp. 189–203, 1965. <https://doi.org/10.1007/BF00186854>

**Кошкін Андрій
Олександрович**

асистент кафедри прикладної математики
Харківський національний університет радіоелектроніки, пр. Науки, 14, м. Харків,
61166, Україна

**Стрельнікова
Олена
Олександрівна**

д.т.н., проф. кафедри прикладної математики
Харківський національний університет радіоелектроніки, пр. Науки, 14, м. Харків,
61166, Україна
провідний науковий співробітник
Інститут енергетичних машин і систем ім. А. М. Підгорного НАН України, вул.
Комунальників, 2/10, м. Харків, 61023, Україна

Аналіз згину багатозв'язних анізотропних плит з пружними включеннями

Актуальність. Визначення напружено-деформованого стану тонких анізотропних плит з інеродними пружними включеннями при поперечному згині є важливою інженерною задачею. Однак загальний випадок плити з декількома довільно розташованими включеннями досі не мав ефективного чисельного або аналітичного розв'язку через значні математичні та обчислювальні труднощі.

Мета. Метою роботи є розробка нового наближеного методу для визначення напруженого стану тонкої анізотропної плити, що містить групу довільно розташованих еліптичних або лінійних пружних включень.

Методи дослідження. Метод базується на застосуванні комплексних потенціалів С. Г. Лехницького. Задача зводиться до визначення функцій узагальнених комплексних змінних для плити-матриці та включень. Ці потенціали представляються відповідними рядами Лорана та поліномами Фабера. Для задоволення контактних граничних умов на контурах включень використовується узагальнений метод найменших квадратів (УМНК). Це зводить задачу до перевизначеної системи лінійних алгебраїчних рівнянь, яка розв'язується за допомогою сингулярного розкладу (SVD).

Результати. Розроблений метод було перевірено шляхом порівняння з відомим точним аналітичним розв'язком для плити з одним еліптичним включенням, що показало повний збіг результатів. Проведено чисельні дослідження впливу відносної жорсткості включень, відстаней між ними та їхніх геометричних характеристик на значення згинальних моментів. Встановлено, що взаємодія між включеннями є суттєвою та призводить до значного зростання моментів при малих відстанях між ними. Ізотропні плити розглядалися як окремий випадок анізотропних.

Висновки. Вперше встановлено, що для лінійних пружних включень особливості моментів, які описуються коефіцієнтами інтенсивності моментів (КИМ), виникають лише у випадках достатньо жорстких або достатньо гнучких включень.

Ключові слова: тонка плита, включення, тріщини, комплексні потенціали, крайова задача, математичне моделювання, чисельні методи, коефіцієнти інтенсивності моментів.

УДК (UDC) 004.85:629.7.05

Lupandin Antonii

*PhD Student, Department of Computer Systems and Robotics;
V. N. Karazin Kharkiv National University, 4 Svobody Square, Kharkiv,
Ukraine, 61022; e-mail: antonii.lupandin@student.karazin.ua;
<https://orcid.org/0009-0002-7591-5152>*

Moroz Olha

*PhD in Computer Science; Associate Professor, Department of Computer
Systems and Robotics;
V. N. Karazin Kharkiv National University, 4 Svobody Square, Kharkiv,
Ukraine, 61022; e-mail: o.moroz@karazin.ua;
<https://orcid.org/0000-0002-4920-4093>*

Analysis of Modern Neural Network Methods for Visual Information Processing in High-Speed UAV Navigation Systems

Relevance. The rapid evolution of Unmanned Aerial Vehicles (UAVs) from remotely piloted systems to fully autonomous high-speed aerial robots has intensified the demand for advanced onboard perception and navigation methods. This need is particularly acute in scenarios where computational latency, sensor noise, and environmental complexity undermine the reliability of classical computer-vision pipelines. Despite recent progress in deep learning, the existing approaches to visual information processing—especially CNN-based detectors, Transformer-based semantic models, and learning-enhanced SLAM modules—remain fragmented and insufficiently adapted to the strict Size, Weight and Power (SWaP) constraints of embedded platforms such as the NVIDIA Jetson series. This motivates a comprehensive analysis of modern neural architectures suitable for real-time, high-velocity UAV operations.

Purpose. The purpose of this study is to analyze state-of-the-art neural network methods for secondary visual processing in UAV navigation systems, compare the applicability of Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs), evaluate their integration into SLAM pipelines, and determine the requirements for hybrid architectures capable of supporting fully autonomous, high-speed flight.

Methods. The research employs a comparative analysis of recent deep-learning approaches, including CNN-based detectors (YOLO family), Transformer-based visual models, deep-learning-enhanced SLAM components, and Deep Reinforcement Learning (DRL) control policies. Evaluation criteria include latency, semantic robustness, dynamic-scene handling, edge-hardware compatibility, quantization performance, pruning potential, and TensorRT optimization efficiency on NVIDIA Jetson devices.

Results. The study establishes that CNNs provide superior real-time performance and remain indispensable for high-frequency reflexive perception, while Vision Transformers offer stronger global context reasoning and robustness to occlusion but suffer from significant computational overhead on embedded GPUs. Deep-learning-based SLAM methods improve feature stability and dynamic-object rejection but require careful integration to maintain real-time constraints. Hardware analysis reveals that quantization, pruning, and TensorRT acceleration are critical for deploying deep models on Jetson-class platforms, although ViTs exhibit limited INT8 quantization tolerance. Based on these findings, the work formulates a conceptual hybrid architecture that combines CNN-driven reflexive processing with Transformer-driven cognitive reasoning.

Conclusions. The results confirm the necessity of developing hybrid neuro-architectures that integrate the speed and hardware efficiency of CNNs with the semantic depth of Transformer-based models. Such architectures represent a promising pathway toward reliable, fully autonomous high-speed UAV navigation. The proposed design principles emphasize hierarchical control, asynchronous perception loops, and hardware-aware optimization as key enablers for next-generation aerial robotic systems.

Keywords: UAV, high-speed navigation, CNN, Vision Transformer, SLAM, Reinforcement Learning, edge computing, Jetson, TensorRT, quantization, pruning, hybrid architectures.

How to quote: A. Lupandin and O. Moroz, “Analysis of Modern Neural Network Methods for Visual Information Processing in UAV Navigation Systems” *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 68, pp. 53-61, 2025. <https://doi.org/10.26565/2304-6201-2025-68-05>

Як цитувати: Lupandin A., Moroz O. Analysis of Modern Neural Network Methods for Visual Information Processing in UAV Navigation Systems. Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління. 2025. вип. 68. С.53-61.

<https://doi.org/10.26565/2304-6201-2025-68-05>

1. Introduction

The operational envelope of Unmanned Aerial Vehicles (UAVs) has expanded dramatically over the last decade. Modern applications—ranging from rapid courier delivery and large-scale environmental monitoring to search-and-rescue operations in disaster zones and dynamic defense maneuvers—now demand a level of autonomy that far exceeds simple GPS waypoint following. As the flight speed of these robotic platforms increases, driven by improvements in rotor efficiency and compact jet-propulsion systems, the allowable margin for error in navigation and obstacle avoidance shrinks to millisecond-level reaction times. This shift has reframed UAV navigation not merely as a geometric estimation task but as a high-velocity perception-and-decision problem [1].

At the heart of this challenge lies the so-called “perception–action loop.” Traditional navigation systems—whether Correlation-Extreme Navigation Systems (CENS) or classical geometric computer-vision pipelines—rely on matching current sensor data to pre-loaded geospatial maps or on extracting handcrafted features such as SIFT or ORB to estimate motion. While mathematically rigorous in controlled or slowly changing environments, these methods exhibit significant brittleness when applied to real-world high-speed flight. They degrade sharply under extreme lighting variation, suffer from the motion blur induced by rapid maneuvers, and fail to generalize when operating in unstructured, semantically complex terrains [2, 3].

To address this gap, the field increasingly turns to the concept of secondary processing—the transformation of raw sensor data (primary information) into a higher-level semantic or structural representation (secondary information). In autonomous flight, this means not simply observing pixels but interpreting them: estimating the UAV’s pose relative to a target or map, detecting and tracking obstacles, recognizing terrain types, and inferring traversability or risk. In essence, secondary processing forms the decision-critical interface between visual perception and control policy [4].

Deep Learning (DL) has become the dominant methodology enabling this transformation. By leveraging large-scale, diverse datasets, neural networks learn feature representations that are robust to noise, illumination variation, high-speed distortions, and other real-world degradations that hinder classical algorithms. Yet, the deployment of such computationally intensive models on Size, Weight, and Power (SWaP) constrained aerial platforms creates a multi-objective optimization problem: accuracy is paramount for safety, while inference latency must remain minimal to maintain stability during aggressive flight.

Against this backdrop, this review synthesizes academic literature published between 2023 and 2025 to map the emerging landscape of neural-network-based UAV navigation. We examine evolving architectural paradigms, including the competition and convergence between CNNs and Vision Transformers as well as the rise of hybrid spatial–temporal models. We also trace how deep learning reshapes Visual SLAM and Visual Odometry, enabling more resilient navigation under sensor degradation. Furthermore, we discuss the rapid shift toward end-to-end Deep Reinforcement Learning (DRL), where perception and control are unified within a single policy network. Finally, we assess the practical realities of deploying these algorithms on embedded edge processors, with particular attention to the NVIDIA Jetson family and comparable low-power accelerators, which increasingly determine what forms of neural autonomy are feasible in high-speed UAV operations [3, 4].

2. Comparative Analysis of Neural Architectures: CNN vs. ViT

The choice of neural architecture is arguably the most consequential design decision in a UAV perception system. It determines not only the theoretical upper bound of visual understanding but also the practical inference dynamics of the model when deployed under stringent Size, Weight, and Power (SWaP) constraints. Today, the field is shaped by the competition—and increasingly, the convergence—of two methodological paradigms: the Convolutional Neural Network (CNN) and the Vision Transformer (ViT).

For more than a decade, CNNs have served as the backbone of visual perception. Their core design principle is the convolution operation, which applies learnable filters over local image regions. This architectural choice embeds strong inductive biases, primarily locality (nearby pixels are correlated) and translation invariance (features retain meaning regardless of spatial position). These properties make CNNs particularly effective in structured environments and for tasks where objects possess distinctive local patterns.

Within UAV-specific applications, the YOLO (You Only Look Once) family of single-stage detectors has become especially prevalent due to its exceptional speed–accuracy trade-off. Unlike two-stage

detectors (e.g., Faster R-CNN), YOLO predicts bounding boxes and class probabilities in a single forward pass, making it highly compatible with the real-time requirements of aerial robotics. The evolution from YOLOv5 to YOLOv8—and more experimental iterations such as YOLOv10/11—has focused on improving the “backbone” and “neck” components. For instance, YOLOv8 introduced C2f modules that enhance gradient propagation and support deeper architectures without the risk of vanishing gradients.

Performance on UAV benchmarks such as VisDrone further underscores CNN effectiveness. Their inductive biases align well with objects that present stable local features (e.g., cars, rooftops, pedestrians), enabling robust detection even under moderate variations in viewpoint or scale. In terms of latency—a critical factor for high-speed flight—lightweight CNNs (e.g., YOLOv8-Nano) achieve real-time performance exceeding 30–60 FPS on embedded hardware like the Jetson Orin Nano. For drones operating at speeds where even a 100 ms delay could result in catastrophic failure, this level of responsiveness is indispensable [5-6].

However, CNNs exhibit structural limitations. Their reliance on local convolutions restricts the ability to capture long-range dependencies early in the network. This becomes problematic in high-altitude or ultra-dynamic UAV imagery, where targets may occupy only a handful of pixels or derive their semantic identity from global scene context (e.g., distinguishing a drone from a bird based on trajectory or surrounding background). Although deeper layers eventually accumulate broader receptive fields, this comes at the cost of spatial resolution due to pooling and downsampling [1].

These shortcomings have spurred interest in Vision Transformers. Originating in natural language processing, Transformers treat data as sequences of tokens. Vision Transformers (ViTs) translate this concept to image processing by dividing the input into patches and applying self-attention across them. This design grants ViTs an immediate global receptive field, enabling every patch to attend to every other patch from the first layer onward.

Recent research – particularly Zhang (2023) – demonstrates the strong potential of ViTs in UAV contexts involving air-to-air detection. In tasks requiring robust recognition of other drones, ViT-based architectures significantly exceeded CNN performance. Notably, Zhang reports that baseline ViTs were approximately $4.6\times$ more robust than comparable CNNs under challenging environmental interference [7]. The ability of self-attention to model global dependencies allows ViTs to suppress background clutter such as clouds or terrain textures, improving detection rates in highly variable scenes. Their resilience also extends to occlusion: because ViTs aggregate information globally, they maintain accuracy even when key object features are partially hidden—a common scenario in urban or forested flight corridors.

Despite their representational strength, ViTs impose substantial computational burdens. Self-attention scales quad

atically, $O(N^2)$, with the number of tokens. For high-resolution aerial imagery –necessary for detecting tiny objects—this leads to excessive memory usage and increased inference times. On edge hardware, pure ViTs often fall below real-time thresholds, achieving only 5–10 FPS on the Jetson Orin Nano [1]. Combined with their weaker inductive biases, which require larger datasets to avoid overfitting [8], pure ViTs remain difficult to deploy in fast-response UAV control loops [7].

These complementary strengths and weaknesses have motivated a shift toward hybrid CNN–Transformer architectures, which seek to integrate the best of both paradigms. In such models, a CNN backbone first extracts low-level spatial features—reducing dimensionality and preserving fine-grained local detail—after which a Transformer module captures global context and long-range dependencies.

Several notable hybrid architectures illustrate this trend:

- LandNet, proposed for 6-DoF camera relocalization, combines spatial convolutions with temporal attention via a Feature Interaction Block. This design enables UAVs to reason about both the geometric structure of landmarks and their configuration across time [9].
- RepEfficientViT extends this idea to edge scenarios by integrating re-parameterized convolutional blocks (RepMBCConv) with lightweight attention. In applications such as agricultural weed recognition—an analogue for detailed ground surveillance—RepEfficientViT surpassed both pure CNNs and ViTs while maintaining an inference latency of ~ 25 ms on a CPU, demonstrating excellent real-time viability [8].
- HCTD (Hybrid CNN–Transformer for Detection) explicitly targets UAV imagery by using convolutions to preserve fine-scale patterns of small objects and self-attention to disentangle complex background clutter characteristic of aerial viewpoints [10].

Across these advances, a clear consensus is emerging: future UAV perception systems will rely not on a single architectural paradigm but on hybrid models that balance local feature extraction, global semantic reasoning, and strict real-time operational constraints.

Table 1 Architectural Trade-offs for UAV Perception

Табл. 1 Архітектурні компроміси у системах сприйняття ПМП

Feature	CNN (e.g., YOLOv8)	Vision Transformer (ViT)	Hybrid (e.g., MobileViT / RepEfficientViT)
Inductive Bias	Strong (Locality/Translation)	Weak (Global Attention)	Balanced
Receptive Field	Local (expands with depth)	Global (all layers)	Local + Global
Small Object Detection	Moderate (loses detail in pooling)	High (preserves context)	High
Inference Latency	Low (< 20ms)	High (> 100ms on edge)	Moderate (20-40ms)
Data Efficiency	High	Low (needs massive data)	Moderate
Edge Optimization	Excellent (TensorRT mature)	Poor (Quantization issues)	Improving

3. Integration of Neural Networks into SLAM and Navigation

Operating in GNSS-denied environments—such as dense urban canyons, forests, tunnels, or areas affected by intentional jamming—requires a UAV to rely on Simultaneous Localization and Mapping (SLAM). Classical Visual SLAM (VSLAM) pipelines, including ORB-SLAM and its derivatives, depend heavily on the detection and tracking of static geometric features. However, high-speed flight introduces severe motion blur, rapid viewpoint transitions, and unpredictable lighting variations, all of which degrade the reliability of handcrafted feature descriptors [1, 11]. As a result, deep learning has begun to permeate SLAM pipelines, not as a wholesale replacement of their mathematically principled foundations, but as a targeted enhancement to the modules most vulnerable to failure.

A major point of fragility in classical SLAM is the feature extraction front-end. Handcrafted features such as ORB, FAST, or BRISK are highly sensitive to illumination changes, blur, and extreme perspective shifts. Neural feature extractors—most prominently SuperPoint and D2-Net—address these weaknesses by learning interest points that remain stable across diverse lighting, scale, and trajectory conditions. When integrated into the SLAM front-end, these extractors significantly improve feature persistence, allowing the UAV to maintain tracking even during aggressive roll, pitch, or acceleration events that would cause classical pipelines to lose localization or trigger a full-system reinitialization [1].

Another foundational assumption of classical SLAM is that the environment is static. In real low-altitude UAV operations, however, the scene is populated with cars, pedestrians, cyclists, and even other drones. These dynamic outliers introduce erroneous correspondences that can corrupt the optimization process, degrade the quality of the map, and induce long-term drift.

To mitigate this, modern SLAM systems increasingly incorporate neural networks for semantic filtering:

- **Semantic Masking:** A neural segmentation network predicts pixel-level semantic classes. Features belonging to “movable” or “dynamic” categories—vehicles, humans, animals—are masked and excluded from the pose estimation and map update stages.
- **Case Study (Luo et al., 2024):** In a recent Multi-Sensor Fusion Dynamic Odometry study, researchers integrated a lightweight neural network into the FAST-LIVO framework to remove dynamic elements before fusing visual data with LiDAR and IMU. This approach substantially reduced trajectory error in dense urban scenes. The network functions as a semantic gatekeeper, ensuring that the SLAM backend optimizes only against reliable, static landmarks [12].

Beyond feature extraction and dynamic filtering, deep learning is increasingly used to enhance sensor fusion—a critical capability for high-speed navigation. Relying on a single modality (e.g., a camera) is risky: a UAV may be blinded by sun glare, fail in low-light conditions, or encounter textureless surfaces

such as glass or snow. Classical fusion pipelines, typically based on Kalman or factor graph frameworks, rely on fixed covariance matrices to weight sensor contributions. However, these static weights cannot adapt to sudden environmental changes.

Neural networks enable adaptive, context-aware fusion:

- **Mechanism:** Instead of relying on hard-coded covariances, a small neural module evaluates the quality of each sensor stream—Visual, LiDAR, IMU—and learns dynamic attention weights. When the visual feed is saturated or motion-blurred, the system automatically up-weights LiDAR or inertial information; when LiDAR becomes unreliable (rain, fog, or low reflectivity surfaces), the visual modality regains prominence.
- **Impact:** This adaptive weighting is crucial for robust state estimation in environments with rapidly changing weather, illumination, and flight dynamics. Neural fusion networks allow UAVs to maintain stable pose estimation even when individual sensors become temporarily unreliable.

Taken together, these advances demonstrate a clear shift in the design of SLAM systems for UAVs: the classical geometric foundations remain intact, but deep learning is increasingly embedded into the perceptual front-ends and fusion layers, transforming SLAM into a hybrid analytical–learning framework that is far more resilient to the operational realities of high-speed flight.

4. Autonomous Navigation via Deep Reinforcement Learning

While SLAM provides the fundamental answer to “Where am I?”, autonomous navigation ultimately depends on the complementary question “How should I move?”. Classical control pipelines—typically built around PID regulators, Linear Quadratic Controllers, or Model Predictive Control (MPC)—treat perception, planning, and actuation as separate sequential modules. Although this modularity offers interpretability and stability, it becomes increasingly restrictive in high-speed scenarios, where UAVs must perform aggressive maneuvers, react to unpredictable obstacles, and adapt to rapidly changing aerodynamics.

Deep Reinforcement Learning (RL) introduces a fundamentally different paradigm: an end-to-end control policy in which a neural network directly maps raw sensor observations (images, depth, inertial states, or fused features) to low-level control commands. Through millions of simulated interactions, the agent optimizes a reward function and gradually learns a policy capable of executing highly non-linear, reflexive, and adaptive behaviors that are extremely difficult—or sometimes impossible—to encode explicitly using model-based controllers.

A compelling demonstration of this capability is presented in Sheng et al. (2024), who investigated UAV motion planning in densely populated, highly dynamic aerial environments. Traditional discrete planners such as A* or sampling-based methods like RRT struggle to replan at sub-50 ms intervals, especially when obstacles move rapidly or unpredictably [1]. RL circumvents this bottleneck by learning a reactive policy that implicitly encodes collision avoidance strategies.

Sheng’s methodology employed an Actor–Critic framework, where the policy network (Actor) generated control commands, and a value network (Critic) estimated long-term expected reward. A carefully engineered reward function balanced three objectives: (1) safety, encouraging large separation from dynamic obstacles; (2) efficiency, rewarding velocity toward the goal; (3) smoothness, penalizing jerk to ensure stable flight dynamics.

During training, the RL agent developed behaviors that resembled reflexes rather than classical planned trajectories. It learned to weave between obstacles, “lead” moving targets, and anticipate collisions before they became imminent. Empirical evaluations showed that the RL approach outperformed Artificial Potential Field (APF) methods in both success rate and mean trajectory quality, especially in densely occupied airspaces.

Beyond obstacle avoidance, RL has demonstrated remarkable potential in controlling systems traditionally considered too chaotic for conventional controllers. One prominent example is slung-load UAV navigation, where a suspended payload introduces non-linear oscillations that couple back into the UAV’s translational dynamics.

In the study by Mohiuddin et al. (2025), RL was applied to this notoriously difficult control problem. The policy learned to regulate thrust in a way that not only stabilized the drone itself but also actively dampened payload oscillations. This holistic, end-to-end learning strategy surpassed hierarchical control approaches, yielding smoother trajectories, improved energy efficiency, and faster point-to-point

transport times [14]. The results highlight a broader trend: RL policies are capable of discovering coordinated strategies that operate simultaneously across multiple dynamic subsystems.

Despite its promise, RL faces a major deployment challenge: the Sim-to-Real (S2R) gap. High-speed UAVs cannot be trained directly in the real world because crashes are expensive and dangerous, while physics engines in simulators (AirSim, Gazebo, Isaac Gym) inevitably introduce discrepancies in aerodynamics, turbulence, sensor noise, and delay.

To mitigate this, modern workflows employ domain randomization, which exposes the agent to a wide distribution of simulated conditions during training. Parameters such as mass, inertia, drag coefficients, wind fields, lighting, motion blur, sensor latency, and texture appearance are randomized within predefined distributions. This forces the policy to learn strategies that generalize across variations, making it robust to the inevitable imperfections of real-world environments [1].

Additionally, researchers increasingly integrate complementary techniques to strengthen real-world deployability:

- Safe RL frameworks impose constraints during learning, limiting the agent’s exploration space to avoid unsafe states while still improving the policy.
- Curriculum learning gradually increases environment complexity—from simple static obstacles to densely dynamic multi-agent scenarios—allowing smoother and more stable convergence.
- Partially Observable RL (PO-MDP) formulations incorporate recurrent networks (LSTMs or Transformers) to compensate for intermittent perception failures, such as temporary camera blindness.
- Multi-Agent RL (MARL) enables cooperative behaviors in UAV swarms, such as coordinated formation flight or distributed obstacle avoidance.

Together, these approaches illustrate a maturing ecosystem around RL for UAVs: the field is transitioning from isolated demonstrations toward practical, reliable deployment of neural policies capable of handling the extreme dynamics of high-speed aerial robots.

5. Hardware Implementation: Edge Computing on NVIDIA Jetson

The advances of Vision Transformers, hybrid architectures, and end-to-end RL policies only become operational when these models can run onboard the UAV. High-speed aerial robots cannot rely on cloud inference due to latency and connectivity and must obey strict limits on power, thermal budget, and memory. In this context, the NVIDIA Jetson family (Nano, Xavier NX, Orin Nano, Orin NX, AGX Orin) has effectively become the standard edge platform for neural perception.

Latency is the key safety constraint. A drone moving at 20 m/s (72 km/h) covers two meters during a 100 ms inference delay—essentially flying “blind” between frames. For high-speed obstacle avoidance, the latency budget is typically under 30 ms, requiring 30–50 FPS depending on the perception stack. These limits directly determine which architectures are viable in real missions.

A comparative study by Meimetis et al. (2025) rigorously evaluates detection models on Jetson-class devices for UAV swarms, clearly contrasting convolutional and attention-based pipelines [15]. Lightweight CNN detectors such as YOLOv8-Nano and YOLOv5-Small consistently achieve >30 FPS on Jetson Orin Nano in FP16, making them suitable for the “reflexive” perception layer responsible for immediate collision avoidance. Transformer-based detectors—DETR variants and heavy ViT backbones—often drop below 10 FPS even on Orin-class hardware because attention operations saturate DRAM bandwidth long before compute throughput becomes the bottleneck [5, 15].

Since raw model performance rarely satisfies onboard real-time constraints, designers increasingly rely on hardware-aware optimizations. NVIDIA TensorRT is central here: it compiles networks into optimized execution graphs via kernel fusion, constant folding, and aggressive memory reuse. Converting a PyTorch-trained YOLO model to a TensorRT engine commonly yields 2–3× speedups on Jetson by reducing global memory traffic and fully exploiting Tensor Cores.

Precision reduction is another widely used technique. Moving from FP32 to mixed-precision FP16 roughly halves memory footprint and often doubles throughput with minimal accuracy loss, making FP16 the practical default for Orin-based UAVs. Pushing to INT8, however, reveals architectural asymmetries. CNNs with relatively well-behaved activations quantize reliably and benefit from substantial acceleration, whereas ViT-style models contain non-Gaussian activations, high-dynamic-range attention logits, and layer normalization, which complicate quantization. Empirical results and developer reports

show that INT8 ViT inference on some Jetson platforms yields little or no gain over FP16 due to quantization/dequantization overheads and the limited availability of highly optimized INT8 attention kernels.

To further reduce model size and energy consumption, structured pruning removes entire convolutional channels, attention heads, or MLP sub-blocks rather than individual weights. For embedded systems this provides two key benefits: (1) the pruned model fits into limited high-speed on-chip memory, reducing DRAM access, and (2) it lowers thermal load, delaying or preventing throttling during sustained high-speed flight. Properly tuned pruning thus delivers real throughput gains without major architectural redesign.

The broader hardware–algorithm co-design trend also includes neural architecture search (NAS) tailored to Jetson devices. These search procedures explore backbones optimized for Tensor Cores, memory locality, and kernel fusion patterns, producing architectures that outperform manually designed counterparts under strict latency constraints. This is particularly promising for hybrid CNN–Transformer models, whose components can be co-optimized for multi-stage execution.

Overall, these optimization strategies highlight a core principle: onboard perception depends not only on neural architecture, but on its interaction with edge hardware. Real-time UAV autonomy emerges from balancing model complexity, parallelism, memory bandwidth, thermal behavior, and numerical precision. As UAVs continue to demand higher speeds and richer perception, hardware-aware optimization will remain central to translating deep learning advances into practical embedded autonomy [1, 5, 8, 15].

Conclusion

The synthesis of contemporary neural methods for UAV navigation reveals a landscape that is both technically mature and inherently heterogeneous. Convolutional Neural Networks continue to dominate real-time perception tasks, providing the latency and stability required for reflexive control during high-speed flight. Vision Transformers, although significantly more powerful in terms of global semantic reasoning and occlusion robustness, remain constrained by their computational overhead on embedded platforms. Reinforcement Learning offers a compelling route toward end-to-end control of nonlinear UAV dynamics, yet it still struggles with generalization and reliable sim-to-real transfer, particularly in safety-critical conditions.

Taken together, these findings indicate that high-speed UAV autonomy cannot rely on any single architectural family. Instead, the most promising trajectory lies in a deliberately hybridized design, where multiple neural paradigms operate synergistically within a hierarchical control framework. In such a system, the first computational tier is responsible for immediate, high-frequency reactions. This reflexive pathway is implemented using aggressively optimized CNNs—such as pruned variants of YOLOv8-Nano—or lightweight hybrid convolution–attention backbones that execute efficiently in FP16 format under TensorRT on Jetson-class hardware. By leveraging the strong inductive biases of convolutions, this pathway can rapidly interpret local visual cues, optical flow patterns, and short-horizon geometric structures, ensuring stable flight and instantaneous obstacle avoidance even under tight latency budgets.

Above this, a slower cognitive pathway provides global situational understanding, long-horizon reasoning, and accumulated map correction. This layer can be realized through Vision Transformers, hybrid attention models, or deep-learning–enhanced SLAM frameworks, all of which excel at capturing wide spatial context and semantic relationships that CNNs inherently struggle to model. Operating at lower frequency (typically 5–10 Hz), it refines the UAV’s understanding of its environment, identifies distant targets, updates global maps, and compensates for the drift or perceptual limitations of the reflexive layer. By separating rapid sensorimotor reflexes from slower cognitive inference—an organization that mirrors biological neural systems—this architectural division offers a principled route to reliable, high-speed autonomy on SWaP-constrained embedded hardware.

Looking ahead, the feasibility of fully autonomous, high-velocity UAVs will depend on how effectively these two pathways can be integrated. Future research must prioritize asynchronous coordination mechanisms that allow low-latency reflexive responses to coexist with slower, more deliberative cognitive reasoning without mutual interference. Equally important is the development of hardware-aware Neural Architecture Search (NAS) methods capable of automatically designing hybrid models that meet strict latency, memory, and energy budgets of edge platforms. Improvements in sim-to-real transfer, robust multi-modal fusion, and safety-oriented reinforcement learning will further strengthen the reliability of such systems.

In conclusion, the path forward lies in embracing hybrid, hierarchical, and hardware-optimized approaches rather than searching for a single universal architecture. By combining the speed of CNN-based reflexes, the contextual capacity of Transformer-driven cognition, and the adaptability of reinforcement learning, next-generation UAVs can achieve the levels of autonomy, robustness, and situational awareness required for safe high-speed flight in complex environments.

REFERENCES

1. Sheng, Y., Liu, H., Li, J., & Han, Q. (2024). UAV autonomous navigation based on deep reinforcement learning in highly dynamic and high-density environments. *Drones*, 8(9), 516. <https://doi.org/10.3390/drones8090516>
2. Scherbinin, V. V., Khusainov, N. S., & Kravchenko, P. P. (2014). Combined correlation-extremal navigation system to identify AV location by terrain relief and landscape objects with the use of the stereo photogrammetry method. *Middle-East Journal of Scientific Research*, 19(4), 479–486. <https://doi.org/10.5829/idosi.mejsr.2014.19.4.13693>
3. Mukhina, M. P., & Seden, I. V. (2014). Analysis of modern correlation extreme navigation systems. *Electronics and Control Systems*, 1(39), 95–101. <https://doi.org/10.18372/1990-5548.39.7343>
4. Sotnikov, A., Tiurina, V., Petrov, K., Lukyanova, V., Lanovyy, O., Onishchenko, Y., Gnusov, Y., Petrov, S., Boichenko, O., & Breus, P. (2024). Using the set of informative features of a binding object to construct a decision function by the system of technical vision when localizing mobile robots. *Eastern-European Journal of Enterprise Technologies*, 3(9(129)), 60–69. <https://doi.org/10.15587/1729-4061.2024.303989>
5. Seeed Studio. (2023, March 30). *YOLOv8 performance benchmarks on NVIDIA Jetson devices*. Seeed Studio Blog. <https://www.seeedstudio.com/blog/2023/03/30/yolov8-performance-benchmarks-on-nvidia-jetson-devices/>
6. D. Du et al. (2019). VisDrone-DET2019: The vision meets drone object detection in image challenge results. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW 2019)* (pp. 213-226). IEEE. <https://doi.org/10.1109/ICCVW.2019.00030>
7. Zhang, J. (2023). Towards a high-performance object detector: Insights from drone detection using ViT and CNN-based deep learning models. In *Proceedings of the 2023 IEEE International Conference on Sensors, Electronics and Computer Engineering (ICSECE)* (pp. 141–147). IEEE. <https://doi.org/10.1109/ICSECE58870.2023.10263514>
8. Liu, T., Wang, Y., Yang, C., Zhang, Y., & Zhang, W. (2025). A lightweight hybrid CNN-ViT network for weed recognition in paddy fields. *Mathematics*, 13(17), 2899. <https://doi.org/10.3390/math13172899>
9. Shen, S., Yu, G., Zhang, L., Yan, Y., & Zhai, Z. (2025). LandNet: Combine CNN and Transformer to Learn Absolute Camera Pose for the Fixed-Wing Aircraft Approach and Landing. *Remote Sensing*, 17(4), 653. <https://doi.org/10.3390/rs17040653>
10. Xue, H., Tang, Z., Xia, Y., Wang, L., & Li, L. (2025). HCTD: A CNN-transformer hybrid for precise object detection in UAV aerial imagery. *Computer Vision and Image Understanding*, 259, 104409. <https://doi.org/10.1016/j.cviu.2025.104409>
11. Favorskaya, M. N. (2023). Deep learning for visual SLAM: The state-of-the-art and future trends. *Electronics*, 12(9), 2006. <https://doi.org/10.3390/electronics12092006>
12. Luo, L., Peng, F., & Dong, L. (2024). Improved multi-sensor fusion dynamic odometry based on neural networks. *Sensors*, 24(19), 6193. <https://doi.org/10.3390/s24196193>
13. Zhu, P., Wen, L., Du, D., Bian, X., Hu, Q., Ling, H., & et al. (2022). Detection and Tracking Meet Drones Challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11), 7380–7399. <https://doi.org/10.1109/TPAMI.2021.3119563>
14. Mohiuddin, M.B., Boiko, I., Tran, V.P. et al. Reinforcement learning for end-to-end UAV slung-load navigation and obstacle avoidance. *Sci Rep* 15, 34621 (2025). <https://doi.org/10.1038/s41598-025-18220-6>
15. Meimetis, D., Daramouskas, I., Patrinooulou, N., Lappas, V., & Kostopoulos, V. (2025). Comparative analysis of object detection models for edge devices in UAV swarms. *Machines*, 13(8), 684. <https://doi.org/10.3390/machines13080684>

Лупандін
Антоній Володимирович

*аспірант кафедри комп'ютерних систем та робототехніки;
Харківський національний університет імені В.Н. Каразіна,
майдан Свободи, 4, Харків, 61022, Україна
e-mail: antonii.lupandin@student.karazin.ua;
<https://orcid.org/0009-0002-7591-5152>*

Мороз
Ольга Юрївна

*PhD комп'ютерних наук; доцент кафедри комп'ютерних систем та
робототехніки;
Харківський національний університет імені В.Н. Каразіна,
майдан Свободи, 4, Харків, 61022, Україна
e-mail: o.moroz@karazin.ua;
<https://orcid.org/0000-0002-4920-4093>*

Аналіз сучасних нейромережевих методів обробки візуальної інформації в системах навігації високошвидкісних ПМР

Актуальність. Стрімка еволюція безпілотних літальних апаратів (БПЛА) — від дистанційно керованих платформ до повністю автономних високошвидкісних повітряних мобільних роботів — зумовлює підвищений запит на вдосконалені методи бортового сприйняття та навігації. Потреба в таких підходах особливо відчутна в умовах, коли обчислювальна затримка, шум сенсорів та складність навколишнього середовища підривають надійність класичних комп'ютерно-зорових систем. Попри суттєвий прогрес у сфері глибокого навчання, наявні підходи до обробки візуальної інформації — зокрема CNN-детектори, семантичні моделі на основі Transformer-архітектур та SLAM-модулі з елементами навчання — залишаються фрагментованими та недостатньо адаптованими до жорстких обмежень за розміром, вагою та енергоспоживанням (SWaP), властивих вбудованим платформам на кшталт NVIDIA Jetson. Це актуалізує потребу в системному огляді сучасних нейроархітектур, придатних для роботи в режимі реального часу на високошвидкісних ПМР.

Мета. Метою дослідження є аналіз сучасних нейронних методів вторинної обробки візуальної інформації в навігаційних системах ПМР, порівняння сфер застосування Convolutional Neural Networks (CNNs) і Vision Transformers (ViTs), оцінювання їх інтеграції у SLAM-підсистеми та визначення вимог до гібридних архітектур, здатних забезпечити повністю автономний високошвидкісний політ.

Методи. У роботі використано порівняльний аналіз сучасних підходів глибокого навчання, включаючи CNN-детектори сімейства YOLO, візуальні моделі на основі Transformer-архітектур, SLAM-компоненти з нейронними модулями та методи управління на основі Deep Reinforcement Learning (DRL). Оцінювання здійснювалося за критеріями затримки, семантичної стійкості, роботи в динамічних сценах, сумісності з вбудованим обладнанням, ефективності квантування, потенціалу структурного проріджування та продуктивності оптимізації TensorRT на пристроях NVIDIA Jetson.

Результати. Дослідження встановило, що CNN-архітектури забезпечують найкращу продуктивність у режимі реального часу та залишаються незамінними для високочастотного рефлекторного сприйняття, тоді як Vision Transformers демонструють кращу здатність до глобального контекстного аналізу й стійкість до оклюзій, але зазнають значних обчислювальних витрат на вбудованих GPU. Нейронно підсилені SLAM-методи покращують стабільність ознак та відслідковування динамічних об'єктів, проте вимагають ретельної інтеграції для збереження роботи в реальному часі. Аналіз апаратної реалізації показав, що квантування, структурне проріджування та оптимізація TensorRT є критично важливими для розгортання глибоких моделей на платформах Jetson, хоча ViT-архітектури демонструють обмежену толерантність до INT8-квантування. На основі отриманих результатів сформульовано концепцію гібридної архітектури, що поєднує рефлекторну швидкодію CNN-модулів із когнітивними можливостями моделей Transformer-типу.

Висновки. Результати дослідження підтверджують необхідність розроблення гібридних нейроархітектур, які інтегрують швидкодію та апаратну ефективність CNN-мереж із семантичною глибиною Transformer-моделей. Такі системи становлять перспективний напрям розвитку надійної, повністю автономної високошвидкісної навігації ПМР. Запропоновані принципи акцентують на ієрархічному управлінні, асинхронних контурах сприйняття та апаратно орієнтованій оптимізації як ключових чинниках для створення ПМР нового покоління.

Ключові слова: ПМР, високошвидкісна навігація, CNN, Vision Transformer, SLAM, Reinforcement Learning, edge-обчислення, Jetson, TensorRT, квантування, проріджування, гібридні архітектури.

УДК (UDC) 004.8

**Omelchenko Ihor
Valeriiovych***PhD student, Department of Mathematical Modeling and Data Analysis
Karazin Kharkiv National University, Svobody Sq 4, Kharkiv, Ukraine,
61022**e-mail: ihor.v.omelchenko@gmail.com;**<https://orcid.org/0009-0007-4474-4916>***Strukov Volodymyr
Mykhailovych***PhD in Technical Sciences, Associate Professor; Head of the Department
of Mathematical Modeling and Data Analysis
Karazin Kharkiv National University, Svobody Sq 4, Kharkiv, Ukraine,
61022**e-mail: volodymyr.strukov@karazin.ua;**<http://orcid.org/0000-0003-4722-3159>*

Reflective memory architecture for adaptive planning in hierarchical LLM agents in virtual environments

Relevance: Large language models (LLMs) can be used as one of the components of autonomous agents that solve sequential decision-making tasks. To improve agent performance, it is necessary to store the history of previous observations and actions, which leads to filling the LLM context window, increasing computational costs, prolonging planning time, and raising memory requirements. A possible approach to addressing this problem is the application of observation reflection methods using LLMs.

Goal: To study the impact of memory reflection methods for autonomous agents based on LLMs. To compare these methods with simpler memory organization approaches.

Research methods: Computational experiments and comparative analysis. Memory organization methods: full episode history, reflection, and reflection with a structured set of memories. The agent performance metrics: task success rate, cumulative reward per episode, and the number of steps required to complete the task.

Results: A memory summarization method based on reflection is proposed for a hierarchical LLM-based agent. The Minigrid ColoredDoorKey environment is used for agent training. Agent code is developed, including components for training the agent in the environment. Computational experiments are conducted to train and evaluate the agent with different memory mechanisms. The performance of different memory mechanisms is evaluated using the following metrics: task completion accuracy, cumulative reward, and the number of steps until episode termination. An analysis and comparison of the results of applying different memory mechanisms to the agent's action planning task in the ColoredDoorKey environment are performed.

Conclusions: The study demonstrates that the use of reflection with a structured set of memories is appropriate for action planning tasks in autonomous agents based on LLMs. The reflection method enables the agent to generalize experience, identify effective rules within large volumes of data with sparse reward signals, and achieve a level of performance comparable to that of a human expert.

Keywords: artificial intelligence, machine learning, deep learning, artificial neural networks, intelligent information systems, automated information systems, natural language processing, large language model, prompt, decision making, agent, memory, virtual environment, Minigrid.

How to quote: I. Omelchenko and V. Strukov, "Reflective memory architecture for adaptive planning in hierarchical LLM agents in virtual environments", *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 68, pp. 62-69, 2025. <https://doi.org/10.26565/2304-6201-2025-68-06>

Як цитувати: Omelchenko I., Strukov V. Reflective memory architecture for adaptive planning in hierarchical LLM agents in virtual environments. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління*. 2025. вип. 68. С.62-69. <https://doi.org/10.26565/2304-6201-2025-68-06>

1 Introduction

Recent advances in the field of large language models (LLMs) have enabled a transition from using models as static text generation tools to creating autonomous agents capable of perceiving the environment, performing actions, and acquiring experience. According to the systematic survey [1], the key difference between an LLM and an LLM-based agent lies in the agent's ability to accumulate knowledge and modify its behavior through experience obtained via long-term interaction with the environment. To enable experience accumulation, LLM-based agents require a memory module that

addresses the inherent limitations of LLMs, namely fixed parameters at inference time and a limited context window length.

The limitation of the finite context window of modern large language models (LLMs) remains a fundamental barrier to their application in tasks that require long-term planning and memory. Although context extension methods exist, simply increasing the input window leads to quadratic growth in computational costs and does not guarantee effective information utilization, especially in long-context scenarios.

Traditional reinforcement learning (RL) methods often require a large number of training samples and computationally expensive fine-tuning of models. An alternative to this approach is the use of In-Context Learning (ICL), where agents learn from their own mistakes in language space through mechanisms of summarization or reflection.

For tasks that require sequential planning, such as navigation in the Minigrad environment, it is important to distinguish between different sources of memory: intra-episode (inside-trial) and inter-episode (cross-trial) memory [1]. Intra-episode memory includes the history of observations, actions, and other messages that arise during the current episode and enables the agent to reason about the current situation. Inter-episode memory contains experience accumulated across multiple trials and serves as the foundation for long-term learning.

Efficient design of inter-episode memory constitutes a key challenge. The following types of memory in LLM-based agents can be distinguished [1]: parametric memory (LLM weights) and textual memory. In this work, we focus on textual memory, as it provides interpretability of decisions and allows manipulation of the agent's knowledge without modifying the LLM weights.

One of the simplest approaches to designing inter-episode memory is storing the complete history of the agent's interaction with the environment. The drawback of this approach is that prolonged interaction leads to overflow of the LLM context window, increased computational complexity, and degradation of the LLM attention mechanism performance. Therefore, modern approaches introduce memory management mechanisms, including summarization, forgetting, and reflection.

Reflection [1] is the process of generating higher-level abstractions, reasoning, and conclusions based on observations. The effectiveness of this approach has been demonstrated in prior work [2-4], where agents formed generalized plans based on key actions from past successful or unsuccessful episodes.

In this paper, we investigate a self-reflective memory mechanism for a hierarchical agent, in which the LLM serves as a high-level planner, while sequences of low-level actions are implemented as procedural options. Building on the Indirect Evaluation methodology described in [1], we conduct an ablation study by comparing agent performance across four configurations, ranging from the complete absence of memory to a comprehensive system with CRUD-like operations over a list of generalized rules. This approach allows us to isolate the impact of the summarization and reflection mechanisms on task performance.

2 Related Works

In the work [4], an agent interacts with a set of tasks and collects a history of successful and unsuccessful attempts to solve them. Subsequently, through reflection, the agent forms a list of rules based on pairs of successful and unsuccessful attempts. The resulting rules are then used by the agent in subsequent attempts to solve the tasks.

In the work [5], a memory mechanism for chatbots is proposed that hierarchically structures memory, storing the complete dialogue history, summarized information about events, and a description of the user's psychological profile. Information summarization is performed periodically, every few episodes. A key feature of this approach is a dynamic memory cleaning mechanism based on the Ebbinghaus forgetting curve. The MemoryBank approach is primarily oriented toward social interaction tasks, such as psychological counseling.

The architecture of Generative Agents was proposed in [2]. A fundamental component of this architecture is the memory stream, a database that stores the agent's complete experience history in natural language form. Unlike approaches that place the agent's experience solely within the LLM context window, the Generative Agents framework employs a dynamic retrieval mechanism that selects relevant entries to construct the context for the current step. The retrieval mechanism accounts for properties of memory entries such as recency, importance, and relevance. The approach also includes a

reflection mechanism that synthesizes low-level observations into high-level abstract conclusions, which are stored in the memory stream alongside low-level observations in a tree-like structure.

The concept of verbal reinforcement learning is introduced in [6]. In this approach, the agent's policy is parameterized not only by the model weights but also by the state of the agent's memory, which contains reflections from previous episodes. The agent's functioning is organized as a loop in which the agent generates a trajectory that is evaluated by an evaluator model; the agent then receives binary or textual feedback, performs reflection, and stores the resulting verbal feedback in long-term memory, enabling behavioral changes in subsequent episodes. The authors note that further improvements to the approach may be achieved by employing more complex memory structures, such as vector databases or SQL-based databases.

LLMs often fail to generate an optimal output on the first attempt. To address this issue, the Self-Refine approach was proposed in [7], which is based on the idea of iterative feedback and refinement. The core idea of this approach is to use an LLM in different roles: a generator, a critic (feedback provider), and a refiner. The agent operates in a loop, where at each iteration an initial solution to the task is generated, feedback on the solution is produced, and the agent refines the previous result based on this feedback. Self-Refine demonstrates the effectiveness of LLMs in generating feedback for themselves and improving their own outputs, achieving significant performance gains without requiring model training. A limitation of this approach is that the feedback is used to improve the current output but is not stored in long-term memory.

The approach proposed in [8] iteratively analyzes the agent's history of previous trajectories and its beliefs, modifying these beliefs to improve the policy. Instead of accumulating episodic memory, this method distills experience into behavioral guidelines and performs world modeling. This enables the agent to form an improved policy that incorporates rules generalized by the agent itself using an LLM.

The authors of [9] propose the MemGPT approach for self-directed memory managing, which simulates an unbounded context length by dynamically moving data between the LLM context and an external storage. In this approach, the LLM not only generates and consumes text but also acts as a controller that manages memory. MemGPT includes a set of functions that allow the agent to add, modify, or retrieve information from external storage based on the current state.

Early approaches to memory organization, such as MemoryBank [5] and Generative Agents [2], rely on fixed memory storage schemas or simple accumulation of history, which limits the agent's ability to generalize knowledge to new environments. A more dynamic and flexible approach, A-MEM, is introduced in [10]. The authors construct agent memory not as a passive storage but as an active, self-organizing system inspired by the Zettelkasten note-taking method. A key feature of A-MEM is the use of a generative approach without fixed schemas. In this approach, the LLM autonomously structures relationships between "notes" (atomic units of knowledge). Despite its flexibility, A-MEM has significant limitations, including high experience recording costs, the absence of a forgetting mechanism, and sensitivity to hyperparameter values.

3 Methods

3.1. Agent Architecture

In this work, we study an agent composed of several modules: a planner, a mediator, and an actor. The planner is built on the LLM Qwen3:8b and performs planning of sequences of abstract actions represented in natural language over the space of procedural options, given the current observation of the environment. The mediator transforms vector-based environment observations into a language representation. The actor receives the abstract plan and executes it by selecting the corresponding options.

Five options were implemented for Minigrid-type environments: *Explore*, *GoTo*, *PickUp*, *Toggle*, and *Drop*. During planning, the agent is restricted to selecting options only from this predefined list. Each option includes a precondition for initiation, an execution procedure that translates the abstract action into a sequence of concrete actions, and a termination condition. The *Explore* and *GoTo* options are implemented using Dijkstra's algorithm and the A* algorithm, respectively.

The options generate linguistic feedback for the planner and provide important information or errors encountered during option execution. The feedback is represented as a tuple consisting of the option execution stage, error status, number of executed steps, and a message. Feedback from the options is used by the planner to correct errors in the plan.

To prevent the agent from entering infinite loops in cases where the planner repeatedly generates the same non-functioning plan, a limit on the number of consecutive identical generated plan was introduced, with a maximum of five repetitions. When this limit is exceeded, the episode is terminated and the task is considered unsuccessful. This modification accelerates the experimental procedure by avoiding waiting for episode termination due to reaching the environment step limit.

3.2. Two-Level Memory System

The agent's interaction with the environment proceeds in episodes. During each episode, the planner selects options, which produce feedback that is stored in memory. The agent's memory is structured by episodes: messages from options are stored sequentially, along with environment observations and the generated plan. At the end of each episode, the memory is augmented with information about the number of steps taken in the episode, task success or failure, and the cumulative reward for the episode.

During agent functioning, the planner's prompt includes a static fragment with the planner's instructions, the task description, a memory fragment corresponding to the current episode, and a set of memory elements formed from the complete history of previous episodes.

We define a *Reflective Memory System* as a dynamic system that evolves in discrete time $t \in N$ through the application of reflection. The state of the memory at certain time t is defined by a set of memory elements, denoted as K_t . Let us denote a set of all agent–environment interaction trajectories at the moment t as $H_t = \{\tau_1, \tau_2, \dots, \tau_t\}$. A mapping from a method-specific subset $\hat{H}_t \subset H_t$ to a set of memory elements K_t that improves the expected reward in future episodes, formally represents the reflection process:

$$K_{t+1} \leftarrow f_{\text{reflect}}(\hat{H}_t, K_t),$$

where f_{reflect} is a function that uses LLM to update the set of memories using selected history trajectories.

The set K_t is defined as a collection of memory elements indexed by unique identifiers. Let $I_t \subset N$ denote the set of active indices at time t . Then, the set of memory elements is defined as:

$$K_t = \{m_k^{(t)}, k \in I_t\},$$

where each memory element $m_k^{(t)}$ is represented as a tuple:

$$m_k^{(t)} = (\phi_k, c_k, \sigma_k),$$

where k is a unique, permanent identifier of the memory element, ϕ_k denotes the category of the memory element, we use single category *RULE*, $c_k \in \Sigma^*$ is the textual content of the memory element (here Σ^* is the set of all possible strings), and $\sigma_k \in [0,1]$ is the importance coefficient of the memory element, initialized as $\sigma_{\text{init}} = 0.5$.

3.3. Reflection and Learning Mechanism

In this study, several approaches to reflect on the history of episodes were studied.

The simplest approach involves pre-collecting the history of episodes, followed by a reflection process using the LLM *Qwen3:30b*. The resulting reflections were added to the planner prompt, and the agent was evaluated on validation and test sets of environments.

The next approach involved sequentially updating the reflections as episodes were completed by the agent. This represents an online update of the planner prompt based on the experience obtained from interactions with the environment.

An enhanced version of reflection was also studied, in which reflections were represented as a list of memories K_t , each assigned an importance score that was updated during agent functioning. This score determines how critical a rule is for the agent's successful performance. A high score prevents the rule from being removed from the memory, serving as a mechanism to mitigate "catastrophic forgetting" during prompt updates.

In this study, the parameters of the LLM were kept fixed. Changes in agent performance were achieved solely by modifying prompts based on the agent's experience obtained from interactions with environment.

The transition from K_t to K_{t+1} occurs through the application of sequence of managing operators to K_t , denoted as a function g_θ . This function uses a LLM to select operators.

The managing function g_θ takes as input a subset of the history and the current memory state:

$$O_t = g_\theta(\hat{H}_t, K_t),$$

where O_t is the sequence of transactional operations on memory generated by the LLM to optimize the memory K_t .

The set of allowed operations is $O = \{\text{CREATE}, \text{UPDATE}, \text{DELETE}, \text{CONFIRM}\}$. Each operation $op \in O_t$ modifies the memory state K_t . We define the transformation function as $T: K \times O \rightarrow K$. Operates for the set O_t are sequentially applied to the memory K_t to produce next state K_{t+1} .

The importance coefficient of a memory element is updated according to an exponential moving function computed from the validation signal received from the LLM-generated transactions *DELETE* or *CONFIRM*:

$$\sigma'_k \leftarrow (1 - \beta)\sigma_k + \beta \cdot \tau_{\text{target}},$$

where τ_{target} is the target importance value of the memory element (0 for penalty, 1 for reinforcement) defined by LLM, and β is a balancing coefficient.

The CREATE operator adds a new element to the set:

$$K'_t \leftarrow K_t \cup (\phi_{\text{new}}, c_{\text{new}}, 0.5).$$

The UPDATE operator is intended to correct erroneous information. The DELETE operator should be used if information in memory is incorrect and needs to be removed. This operator functions as a soft delete by lowering the importance coefficient of the element; when the coefficient falls below a critical threshold, the element is permanently removed at the memory pruning stage. The CONFIRM operator increases the importance coefficient of the memory element.

After applying all selected LLM-generated operations O_t , a pruning stage occurs, in which memory elements with importance coefficients below $\epsilon = 0.1$ are removed.

3.4. Experimental Design

In this study, the Minigrad ColoredDoorKey environment was used, which is a two-dimensional, discrete-state and discrete-action environment. The environment presents a task requiring the agent to open locked doors using a key of the matching color. At the beginning of an episode, objects in the environment are hidden, and the agent must first explore the environment. The environment contains two keys of different colors, one of which is required to open the door, while the another serves as a distractor. Due to limited visibility and incomplete information, the agent must plan actions to obtain information about the door color, find the matching key, pick it up, navigate to the door, and open it.

Two LLMs were used in the experiment: Qwen3:8b and Qwen3:30b. The larger model, Qwen3:30b, was utilized to reflect upon episode histories and managing memory. The smaller model, Qwen3:8b, was used in the planning module, leveraging the reflections produced by the larger model to plan in the space of options. Using models of different sizes is advantageous because planning must be fast, as it is executed multiple times per episode. In contrast, history reflection occurs once every few episodes, and thus weaker constraints on runtime are acceptable for this stage.

To evaluate the contribution of the proposed memory mechanism, ablation studies were conducted. The simplest variant is a memoryless agent, which receives only direct environment observations and has no access to either the current episode history or the overall history. This approach provides a lower bound on the accuracy achievable by the agent. The next level of complexity is episodic memory, in which the agent receives the history of the current episode. This allows the agent to correct mistakes made during the episode but does not enable knowledge transfer across episodes. A more advanced approach allows the agent to access the memories of the last N episodes, simulating long-term memory, but this fills the LLM context and dilutes the agent's attention.

The agent-environment interaction was divided into three stages: Train, Eval, and Test. For each stage, a separate fixed set of random seeds was selected to ensure reproducibility. During the Train stage, the agent collected interaction history with the environment, which was then used either for forming offline reflections or for online reflections updates. The Eval stage was executed in parallel with Train every N episodes to assess the dynamics of agent performance. The Test stage served as the final evaluation, in which the agent was tested over a large number of episodes to compute the final performance metrics.

3.5. Evaluation Metrics

The following metrics were used to evaluate agent performance. The primary performance metric is the *success rate*, calculated as the average task completion success of the agent per episode. *Return* is computed as the average of the sum of rewards throughout the episode, taking into account a penalty for the number of steps taken. *Steps per Episode* measures the number of steps the agent took to complete the episode (successfully or unsuccessfully). Number of Re-plans counts the number of times the

planner had to regenerate a plan to produce a working plan. Planning Duration measures the time spent in the planning stage, used to evaluate the increase in computational cost as the agent's memory size grows.

4 Experiments

We conducted computational experiments in the Minigrid ColoredDoorKey environment. All experiments were performed on a computer system with an NVIDIA RTX 3090 GPU (24 GB VRAM). The LLM was run using the Ollama 0.13.5 utility. We used LLMs quantized in the Q4_K_M format. The "temperature" hyperparameter was set to 0.1. The maximum number of generated tokens was limited to 4096.

We compared several approaches to agent memory organization:

1. **Without Memory:** The agent had no memory or instructions.
2. **Baseline E-1:** The agent remembered only the current episode.
3. **Offline Reflection:** The agent remembered the current episode and received instructions from LLM Qwen3:30b, generated in advance on pre-collected data.
4. **Text Reflection E-5:** The agent had memory of the current episode and performed unstructured reflection every 5 episodes on the last five episodes.
5. **Text Reflection E-5 CP-1:** The agent had memory of the current episode and performed unstructured reflection every 5 episodes on the last pair of successful and unsuccessful episodes.
6. **Text Reflection E-5 CP-3:** The agent had memory of the current episode and performed unstructured reflection every 5 episodes on the last three pairs of successful and unsuccessful episodes.
7. **Reflection E-2:** The agent had memory of the current episode and performed structured reflection with managing operators every 2 episodes.
8. **Reflection E-5:** The agent had memory of the current episode and performed structured reflection with managing operators every 5 episodes.
9. **Manual Instructions:** The agent had memory of the current episode and received high-quality instructions formulated by a human-expert.

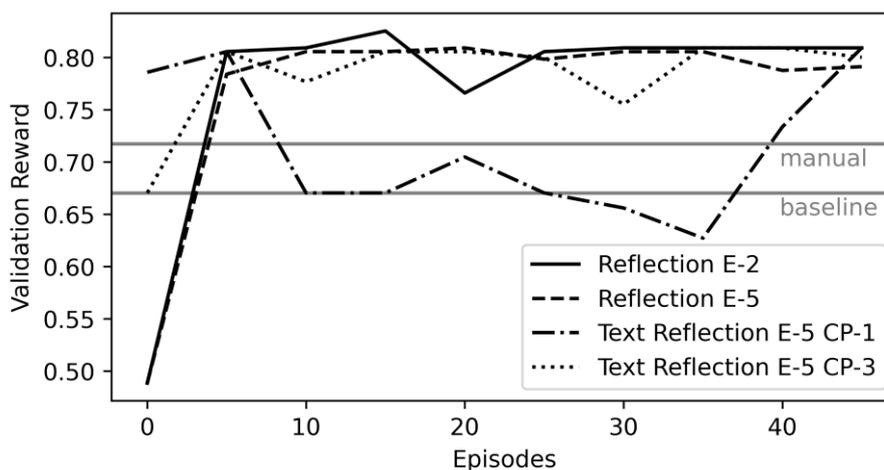


Рис. 1 Залежність сумарної нагороди на валідаційному наборі від епізоду під час навчання

Fig. 1 Dependency of cumulative reward on the validation set as a function of episode during training

Figure 1 shows the dynamics of cumulative reward on the validation set of environments. It can be observed that the cumulative reward for memory with structured reflection (Reflection) reaches maximum values within the first 4 iterations of reflection and achieves better performance than the agent with rules written by a human expert.

An example of the final rule set for the Reflection method:

Model Output

0: Before picking up or toggling an object, move to it to ensure correct interaction, verify its presence, color match, and that the key is held before attempting to toggle. If holding a key and needing to pick up another, drop the current key first. Only move to an object if it is visible or known to be present. Verify key color matches door color before toggling. Move to the object even if visible but not in front to ensure correct targeting.

1: After picking up the key, explore for the door to locate it before attempting to open it.

2: Before toggling the door, verify that the key color matches the door color, move to the door, and ensure the key is held before attempting to toggle.

Табл. 1 Порівняння різних підходів до організації пам'яті агента

Table 1 Comparison of different approaches to agent memory organization

Method	Accuracy	Reward	Steps
Without Memory	0.68	0.4994	52.06
Baseline E-1	0.65	0.4896	52.82
Offline Reflection	0.72	0.5298	48.02
Text Reflection E-5	0.74	0.5470	47.44
Text Reflection E-5 CP-1	0.83	0.6215	40.16
Text Reflection E-5 CP-3	0.96	0.7173	30.96
Reflection E-2	0.99	0.7353	29.29
Reflection E-5	0.98	0.7345	29.27
Manual Instructions	0.99	0.7345	29.62

Table 1 presents a comparison of metric values for agents with different memory mechanisms. The lowest performance is observed for the baseline, corresponding to an agent without long-term memory. As the memory complexity increases, agent performance improves. The most advanced method, Reflexion — structured reflection with a list of memories and managing operators — demonstrates the best results, comparable to or better than the agent with rules provided by a human expert.

6 Conclusions

In this study, the following was accomplished. We proposed using In-Context Learning with reflection via LLM for agent training in virtual environments. The Minigrid ColoredDoorKey environment was prepared for agent training. Agent training code was implemented for the Minigrid ColoredDoorKey environment. Computational experiments were conducted to train and test the agent with different memory mechanisms. The performance of various memory mechanisms was evaluated based on metrics: task completion accuracy, cumulative reward, and number of steps per episode. An analysis and comparison of the influence of memory mechanisms on agent action planning in the ColoredDoorKey environment was performed.

The results indicate that using the reflection method via LLM in an autonomous agent improves agent performance during interaction with a virtual environment. Task completion accuracy shows that the reflective agent can discover rules enabling it to reach human-expert-level accuracy. Learning curves demonstrate that maximum accuracy is achieved by the third reflection step. Additionally, the use of reflection reduces the number of steps required to complete a task and increases the cumulative reward obtained by the agent over the course of an episode.

REFERENCES

1. Zhang Z., Dai Q., Bo X. et. al. A survey on the memory mechanism of large language model-based agents. *ACM Transactions on Information Systems*. 2025. Vol. 43. P. C. 1—47.

2. Park J., O'Brien J., Cai C. et. al. «Generative agents: Interactive simulacra of human behavior». In: Proceedings of the 36th annual acm symposium on user interface software and technology. 2023, c. 1-22.
3. Zhu X., Chen Y., Tian H. et. al. Ghost in the Minecraft: Generally Capable Agents for Open-World Environments via Large Language Models with Text-based Knowledge and Memory. 2023. arXiv: 2305.17144 [cs.AI]. URL: <https://arxiv.org/abs/2305.17144>.
4. Zhao A., Huang D., Xu Q. et. al. «Expel: Llm agents are experiential learners». In: Proceedings of the AAAI Conference on Artificial Intelligence. Т. 38. 17. 2024, c. 19632—19642.
5. Zhong W., Guo L., Gao Q. et. al. «Memorybank: Enhancing large language models with long-term memory». In: Proceedings of the AAAI Conference on Artificial Intelligence. Т. 38. 17. 2024, c. 19724-19731.
6. Shinn N., Cassano F., Berman E. et. al. Reflexion: Language Agents with Verbal Reinforcement Learning. 2023. arXiv: 2303.11366 [cs.AI]. URL: <https://arxiv.org/abs/2303.11366>
7. Madaan A., Tandon N., Gupta P. та ін. Self-Refine: Iterative Refinement with Self-Feedback. 2023. arXiv: 2303.17651 [cs.CL]. URL: <https://arxiv.org/abs/2303.17651> .
8. Zhang W., Tang K., Wu H. та ін. Agent-Pro: Learning to Evolve via Policy-Level Reflection and Optimization. 2024. arXiv: 2402.17574 [cs.AI]. URL: <https://arxiv.org/abs/2402.17574> .
9. Packer C., Wooders S., Lin K. та ін. MemGPT: Towards LLMs as Operating Systems. 2024. arXiv: 2310.08560 [cs.AI]. URL: <https://arxiv.org/abs/2310.08560> .
10. Xu W., Liang Z., Mei K. та ін. A-MEM: Agentic Memory for LLM Agents. 2025. arXiv: 2502.12110 [cs.CL]. URL: <https://arxiv.org/abs/2502.12110> .

Омельченко Ігор Валерійович *Аспірант, кафедра математичного моделювання та аналізу даних Харківський національний університет ім. В.Н. Каразіна. Площа Незалежності, 4, Харків, Харківська область, 61022*

Струков Володимир Михайлович *к.т.н., доцент; завідувач кафедри математичного моделювання та аналізу даних Харківський національний університет ім. В.Н. Каразіна. Площа Незалежності, 4, Харків, Харківська область, 61022*

Архітектура рефлексивної пам'яті для адаптивного планування ієрархічних LLM-агентів у віртуальних середовищах

Актуальність. Великі мовні моделі (LLM) можуть бути використані як один з елементів автономних агентів, що вирішують завдання послідовного прийняття рішень. Для покращення роботи агентів потрібно зберігати історію попередніх спостережень та дій, що призводить до заповнення контекстного вікна LLM, збільшення кількості обчислень, тривалості планування та підвищує вимоги до пам'яті. Можливий підхід до вирішення цієї проблеми полягає у застосуванні методів узагальнення спостережень з використанням LLM.

Мета. Дослідити вплив методів узагальнення пам'яті автономних агентів на основі LLM та рефлексії. Порівняти з простішими методами організації пам'яті.

Методи дослідження. Методи дослідження: обчислювальний експеримент, порівняльний аналіз. Методи організації пам'яті: повна історія епізодів, рефлексія, рефлексія зі структурованим набором правил. Використані метрики якості роботи агента: успішність вирішення завдання, сумарна нагорода за епізод, кількість кроків для вирішення завдання.

Результати. Запропоновано використовувати метод узагальнення пам'яті шляхом рефлексії для ієрархічного агента на основі LLM. Розглянуто середовище Minigrid ColoredDoorKey для навчання агента. Створено код агента, зокрема для навчання агента в середовищі. Проведено обчислювальні експерименти з навчання та тестування агента з різними механізмами пам'яті. Проведено оцінку якості роботи різних механізмів пам'яті на основі метрик: точність виконання завдання, сумарна нагорода, кількість кроків до завершення епізоду. Виконано аналіз та порівняння результатів застосування механізмів пам'яті до задачі планування дій агентом в середовищі ColoredDoorKey.

Висновки. Дослідження демонструє, що застосування методу рефлексії з структурованим набором правил є доцільним в задачах планування дій автономними агентами з використанням LLM. Метод рефлексії дозволяє узагальнювати досвід агента, знаходити ефективні правила в значному об'ємі даних з розрідженим сигналом нагороди та досягати рівня ефективності порівняного з людиною-експертом.

Ключові слова: штучний інтелект, машинне навчання, глибоке навчання, штучні нейронні мережі, інтелектуальні інформаційні системи, автоматизовані інформаційні системи, обробка природної мови, велика мовна модель, промпт, прийняття рішень, агент, пам'ять, віртуальне середовище, Minigrid.

УДК (UDC) 004.77:620.9

**Стецюренко Ілля
Станіславович**

магістрант кафедри системного проектування і спеціалізованих комп'ютерних систем факультету прикладної математики, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського» Україна, 03056, м. Київ, пр. Берестейський, 37
e-mail: stetsiurenko.illia@ill.kpi.ua;
<https://orcid.org/0009-0009-8453-2154>

**Петрашенко Андрій
Васильович**

кандидат технічних наук, доцент кафедри системного проектування і спеціалізованих комп'ютерних систем факультету прикладної математики, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського» Україна, 03056, м. Київ, пр. Берестейський, 37
e-mail: petrashenko@gmail.com;
<https://orcid.org/0000-0003-0239-1706>

Спосіб оптимізації енергоживлення IoT-системи моніторингу кліматичних показників на основі адаптивних алгоритмів

Актуальність. Стрімке зростання кількості пристроїв Інтернету речей (IoT), яке за прогнозами сягне 29 мільярдів до 2030 року, ставить перед розробниками нові виклики. Значна частина таких пристроїв — це автономні сенсорні вузли, що використовуються в агросекторі, екології та інфраструктурі «розумних міст». Розміщення у важкодоступних місцях робить їх залежними від автономних джерел живлення. Проблема полягає в тому, що традиційні алгоритми роботи, які базуються на фіксованих інтервалах передачі даних, є енергетично неефективними: вони витрачають ресурс батареї на передачу надлишкових даних у стабільних умовах або не забезпечують достатньої дискретизації при різких змінах параметрів середовища.

Мета. Розробка способу оптимізації енергоживлення IoT-системи моніторингу кліматичних показників на основі адаптивних алгоритмів та проведення порівняльного аналізу енергоефективності різних комунікаційних архітектур (Wi-Fi, BLE, LoRaWAN) для виявлення оптимальних рішень.

Методи дослідження. Для досягнення мети використано експериментально-аналітичний підхід. В якості апаратної платформи обрано мікроконтролер ESP32 (модуль ESP-WROOM-32), який поєднує високу продуктивність, наявність бездротових інтерфейсів та розвинені режими енергозбереження (Deep Sleep, ULP). Збір кліматичних даних здійснюється сенсором Bosch BME680. Хмарна інфраструктура реалізована на базі Amazon Web Services (AWS) з використанням безсерверної архітектури (AWS Lambda, DynamoDB, Redis). Для оцінки енергоефективності розроблено чотири еволюційні версії програмного забезпечення: від базової (активний режим) до адаптивних (локальна та хмарна адаптація). Ключовою метрикою ефективності обрано середній струм споживання за повний робочий цикл.

Результати. Розроблено гібридну модель керування енергоспоживанням, реалізовану у двох парадигмах: локальна адаптація (на пристрої) та хмарна адаптація (на сервері). Проведено серію натурних вимірювань для протоколів HTTP, MQTT, CoAP, BLE та LoRaWAN. Встановлено, що перехід на протокол CoAP з адаптивним алгоритмом у мережі Wi-Fi дозволяє знизити середнє споживання струму з 60.46 мА (базовий режим) до 12.47 мА (економія ~79%). Для архітектури LoRaWAN досягнуто зниження з 96.78 мА до 12.63 мА (економія 87%). Виявлено неефективність хмарної адаптації для протоколу MQTT через затримки зв'язку.

Висновки. Інтеграція адаптивних алгоритмів, що динамічно керують тривалістю фази сну, дозволяє знизити енергоспоживання на 70-87% порівняно з базовими режимами. Для систем з інфраструктурою Wi-Fi найбільш енергоефективним є протокол CoAP. Для задач, що вимагають максимальної автономності та радіусу дії, оптимальним вибором є LoRaWAN з локальним адаптивним алгоритмом.

Ключові слова: Інтернет речей, енергоефективність, адаптивні алгоритми, ESP32, Deep Sleep, MQTT, CoAP, LoRaWAN, BLE.

Як цитувати: Стецюренко І. С., Петрашенко А. В. Спосіб оптимізації енергоживлення IoT-системи моніторингу кліматичних показників на основі адаптивних алгоритмів. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління*. 2025. вип. 68. С.70-76. <https://doi.org/10.26565/2304-6201-2025-68-07>

How to quote: Stetsiurenko, I. S., & Petrashenko, A. V. "Method of Power Supply Optimization for IoT Climate Monitoring System Based on Adaptive Algorithms", *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 68, pp. 70-76, 2025. <https://doi.org/10.26565/2304-6201-2025-68-07> [in Ukrainian]

1. Вступ

Інтернет речей (IoT) трансформувався з концептуальної парадигми в глобальну технологічну реальність. За прогнозами, кількість підключених пристроїв продовжить експоненційне зростання, охоплюючи критично важливі сфери: від точного землеробства до моніторингу кліматичних змін та управління міською інфраструктурою. Специфіка цих застосувань часто вимагає розгортання сенсорних вузлів у віддалених локаціях, де єдиним джерелом енергії є акумуляторні батареї. У таких умовах енергоефективність стає ключовим фактором, що визначає вартість володіння (TCO) та надійність системи. Традиційний підхід до передачі даних за фіксованим розкладом (static scheduling) демонструє свою неефективність: значна частина енергії витрачається на передачу надлишкових даних у періоди стабільності параметрів, тоді як фіксований інтервал може бути недостатнім для фіксації швидких динамічних змін [1].

2. Аналіз останніх досліджень і публікацій

Проблема оптимізації енергоспоживання в IoT є предметом багатьох сучасних досліджень. Значна увага приділяється апаратним методам, зокрема використанню режимів глибокого сну (Deep Sleep) у сучасних мікроконтролерах, таких як ESP32 [2]. Інший напрямок досліджень фокусується на виборі ефективних протоколів передачі даних. У роботах [3, 4] проведено порівняльний аналіз протоколів прикладного рівня (MQTT, CoAP, HTTP) і показано переваги CoAP для обмежених мереж. Також активно досліджуються технології LPWAN, зокрема LoRaWAN, як рішення для далекого зв'язку з низьким споживанням [5,6]. Проте в існуючій літературі бракує комплексних досліджень, які б поєднували апаратну оптимізацію, вибір протоколу та адаптивні алгоритми керування в рамках єдиної системи, а також надавали пряме експериментальне порівняння різних архітектур (Wi-Fi, BLE, LoRaWAN) для виконання ідентичної задачі моніторингу.

3. Постановка задачі

У даній роботі вирішується задача підвищення енергоефективності IoT-системи моніторингу кліматичних показників. Необхідно розробити метод адаптивного керування режимами роботи пристрою та експериментально визначити найбільш ефективну архітектуру зв'язку, що забезпечить баланс між актуальністю даних та часом автономної роботи.

4. Виклад основного матеріалу

4.1. Архітектура та апаратне забезпечення системи

Для реалізації енергоефективної системи моніторингу було застосовано експериментально-аналітичний підхід. В якості центрального обчислювального вузла обрано мікроконтролер ESP32 (модуль ESP-WROOM-32) [2]. Вибір цієї платформи зумовлений наявністю унікального для даного класу пристроїв ULP-співпроцесора (Ultra Low Power coprocessor). Це дозволяє реалізувати дворівневу архітектуру обробки даних:

- Основні ядра (Xtensa LX6): Виконують ресурсоємні задачі, такі як підключення до Wi-Fi, SSL/TLS шифрування та формування JSON-пакетів. Споживання струму в активному режимі сягає 190–240 мА.

- ULP-співпроцесор: Здатний періодично опитувати сенсори та виконувати прості логічні операції, поки основні ядра знаходяться в режимі глибокого сну (Deep Sleep). Споживання при цьому знижується до рівня ~150 мкА. У розробленій системі ULP використовується для моніторингу критичних відхилень параметрів, що дозволяє не будити основний процесор, якщо показники знаходяться в межах норми.

Для збору кліматичних даних (температура, вологість, тиск, якість повітря) використано інтегрований сенсор Bosch BME680 [7]. Критичним аспектом енергоспоживання цього сенсора є вимірювання летких органічних сполук (VOC), яке вимагає нагрівання чутливої пластини до 320°C. Це створює пікове навантаження на рівні 12–18 мА. Для оптимізації в алгоритмі передбачено адаптивне керування профілем нагріву: у режимах економії енергії вимірювання VOC виконується з меншою частотою, ніж вимірювання базових параметрів (Т/В/Р), що суттєво знижує середній струм споживання.

Для дослідження архітектури глобальних мереж (LPWAN) стенд оснащено трансивером Semtech SX1276 [6], який підключається через інтерфейс SPI. Це дозволяє реалізувати та порівняти роботу пристрою в мережі LoRaWAN.

4.2. Архітектура та апаратне забезпечення системи

В основу програмної реалізації покладено модель скінченного автомата (Finite State Machine — FSM). Система може перебувати в одному з чотирьох дискретних станів:

1. STATE_SLEEP: Мікроконтролер перебуває в режимі Deep Sleep. Активними залишаються лише RTC-таймер та енергонезалежна пам'ять RTC. Струм споживання мінімальний ($I_{\text{снy}} \approx 10 \dots 20 \text{ мА}$).
2. STATE_WAKEUP: Пробудження, ініціалізація шини I2C, зчитування даних із сенсора BME680.
3. STATE_ANALYZE: Локальний аналіз даних. Система порівнює поточні значення V_{cur} із попередніми V_{prev} , збереженими в пам'яті RTC. Розраховується абсолютна дельта змін $\Delta = [V_{\text{cur}} - V_{\text{prev}}]$
4. STATE_TRANSMIT: Активація радіомодуля (Wi-Fi/BLE/LoRa), встановлення з'єднання, передача даних, отримання підтвердження або команди.

Ключовим елементом розробленого методу є алгоритм адаптації інтервалу сну ($T_{\text{снy}}$). На відміну від статичного підходу, де $T_{\text{снy}} = \text{const}$, у запропонованому методі тривалість сну є функцією від стабільності середовища. Логіка адаптації описується наступним чином:

Якщо $\Delta < \delta_{\text{thr}}$ (де δ_{thr} - порогове значення стабільності), система переходить у режим енергозбереження, експоненційно збільшуючи інтервал сну для наступного циклу:

$$T_{\text{снy}}^{\text{new}} = \min(T_{\text{снy}}^{\text{old}} * k_{\text{inc}}, T_{\text{max}}) \quad (1)$$

де k_{inc} — коефіцієнт збільшення (наприклад, 1.5 або 2.0), а T_{max} — максимально допустимий інтервал між передачами (heartbeat).

Якщо $\Delta \geq \delta_{\text{thr}}$, система фіксує подію зміни стану, передає дані та скидає інтервал до мінімального значення для детального моніторингу динаміки процесу:

$$T_{\text{снy}}^{\text{new}} = T_{\text{min}} \quad (2)$$

Такий підхід дозволяє уникнути передачі надлишкових даних (data redundancy), коли параметри середовища не змінюються, зберігаючи при цьому високу частоту дискретизації при виникненні аномалій. Алгоритм реалізовано у двох парадигмах: локальна адаптація (рішення приймає мікроконтролер) та хмарна адаптація (рішення приймає сервер AWS Lambda на основі історичних трендів).

4.3. Аналіз особливостей реалізації комунікаційних стеків

Значний вплив на енергоефективність автономного пристрою мають накладні витрати (overhead) протоколів передачі даних. У роботі досліджено три основні підходи для IP-мереж (Wi-Fi):

- HTTP (REST): Характеризується найбільшими накладними витратами. Кожна транзакція вимагає встановлення нового TCP-з'єднання (3-way handshake), встановлення TLS-сесії (обмін сертифікатами) та передачі текстових заголовків. Це призводить до значного часу перебування в активному режимі ($T_{\text{акт}} > 4 \text{ с}$) навіть для передачі кількох байт корисного навантаження.
- MQTT: Використовує постійне TCP-з'єднання. Однак, у режимі Deep Sleep радіомодуль вимикається, і з'єднання розривається. Тому при кожному пробудженні пристрій змушений проходити процедуру CONNECT заново, що нівелює переваги легкості протоколу [8]. Крім того, у режимі v4 (Cloud Adaptation) MQTT показав найгірші результати через свою асинхронну природу: пристрій змушений утримувати радіоканал відкритим, очікуючи на PUBLISH повідомлення від сервера з новим часом сну, що збільшує активний час до 7–10 секунд.
- CoAP (Constrained Application Protocol): Працює поверх протоколу UDP [9]. Відсутність важкого рукостискання TCP та компактний бінарний заголовок (4 байти) дозволяють мінімізувати час перебування в ефірі [10]. Для забезпечення надійності використовується механізм підтверджуваних повідомлень (Confirmable messages), а безпека реалізується через DTLS. Експерименти показали, що CoAP забезпечує найменший час активності ($T_{\text{акт}} \approx 1,4 \text{ с}$), що робить його оптимальним для Wi-Fi мереж.

4.4. Специфіка реалізації LoRaWAN

Для архітектури глобальної мережі використано стек LoRaWAN класу А. Пристрої цього класу є найбільш енергоефективними, оскільки вони ініціюють зв'язок самостійно (Uplink) і

відкривають короткі вікна прийому (RX1, RX2) лише після передачі. Це накладає обмеження на реалізацію хмарної адаптації (v4): сервер може надіслати команду зміни інтервалу сну лише у відповідь на повідомлення від пристрою. Якщо серверна інфраструктура не встигає сформулювати відповідь у межах жорстких таймінгів RX-вікон (1-2 секунди), команда буде відкладена до наступного сеансу зв'язку.

Експериментально встановлено, що для LoRaWAN оптимальним є використання локальної адаптації (v3). Це виключає необхідність очікування downlink-повідомлень, дозволяючи пристрою переходити в сон одразу після завершення передачі (TX Done).

4.5. Методика експериментального дослідження

Для верифікації запропонованих моделей розроблено чотири еволюційні версії програмного забезпечення:

v1 (Baseline): Базова версія без оптимізації. Пристрій постійно активний.

v2 (Deep Sleep): Циклічний сон з фіксованим інтервалом (наприклад, 30 с).

v3 (Local Adaptation): Адаптивний алгоритм з прийняттям рішень на пристрої.

v4 (Cloud Adaptation): Адаптивний алгоритм з прийняттям рішень у хмарі (AWS Lambda).

Ключовою метрикою ефективності обрано середній струм споживання ($I_{сер}$) за повний робочий цикл, який формулюю:

$$I_{сер} = \frac{I_{акт} * T_{акт} + I_{сну} * T_{сну}}{T_{акт} + T_{сну}} \quad (3)$$

де $I_{акт}$ та $T_{акт}$ - струм і тривалість активної фази, а $I_{сну}$ та $T_{сну}$ - струм і тривалість фази сну. Дана методика дозволяє об'єктивно порівняти ефективність різних архітектур незалежно від ємності джерела живлення.

5. Результати експерименту

В ході експериментального дослідження було проведено серію натурних вимірювань енергоспоживання для п'яти комунікаційних протоколів: HTTP, MQTT, CoAP (у мережі Wi-Fi), а також BLE та LoRaWAN. Для кожного протоколу (де це архітектурно можливо) було протестовано чотири режими роботи: від базового (v1) до хмарно-адаптивного (v4). Результати вимірювань усередненого струму споживання ($I_{сер}$) та тривалості активної фази $T_{акт}$ наведено в таблиці 1.

Протокол	Режим	Такт (с)	Iсер (мА)	Зниження відн. v1
HTTP	v1	4.33	59.61	0%
	v2	8.96	24.68	58.6%
	v3	8.78	17.17	71.2%
	v4	5.05	16.59	72.2%
MQTT	v1	7.39	63.43	0%
	v2	5.69	22.08	65.2%
	v3	5.62	16.11	74.6%
	v4	10.43	15.34	75.8%
CoAP	v1	2.63	60.46	0%
	v2	4.54	19.89	67.1%
	v3	4.46	12.61	79.2%
	v4	4.49	12.47	79.4%
BLE	v1	2.98	56.88	0%
	v2	6.93	22.43	60.6%
	v3	3.56	14.38	74.7%
LoRaWAN	v1	3.62	96.78	0%
	v2	3.32	19.20	80.2%
	v3	2.79	12.63	87.0%

Аналіз отриманих експериментальних даних дозволяє стверджувати, що впровадження режиму глибокого сну є найбільш вагомим фактором оптимізації, оскільки перехід від базового режиму постійної активності до режиму циклічного сну забезпечує зниження середнього енергоспоживання на 58–80% залежно від обраного протоколу. Найбільший відносний ефект спостерігається для архітектури LoRaWAN, де високе базове споживання трансивера в режимі очікування повністю нівелюється при переході системи в сон, що дозволяє досягти зниження середнього струму більш ніж на 80%. Подальше застосування розроблених адаптивних алгоритмів дозволило досягти додаткової економії енергоресурсів за рахунок динамічного збільшення інтервалів сну в періоди стабільності вимірюваних параметрів. Для протоколів сімейства IP це забезпечило зниження середнього струму до 12–17 мА, що становить сумарну економію до 79% відносно базового рівня. Важливо зазначити, що різниця в ефективності між локальною та хмарною моделями адаптації виявилася статистично незначною, що підтверджує гіпотезу про те, що ключовим фактором енергозбереження є саме адаптивне керування шпаруватістю передачі даних, а не локалізація обчислювальних процесів.

Порівняльний аналіз протоколів передачі даних у мережі Wi-Fi виявив суттєву перевагу протоколу CoAP, який продемонстрував найнижчий показник середнього струму споживання на рівні 12,47 мА. Це пояснюється використанням транспортного протоколу UDP, що дозволяє уникнути накладних витрат на встановлення з'єднання та процедури рукоштовкування, характерних для TCP, і тим самим мінімізувати час перебування радіомодуля в активному стані. У той же час протокол MQTT у режимі хмарної адаптації продемонстрував зниження ефективності через збільшення тривалості активної фази, що зумовлено необхідністю утримання з'єднання для очікування асинхронної команди від сервера. Результати дослідження альтернативних архітектур показали, що технологія LoRaWAN у режимі локальної адаптації забезпечує енергоефективність на рівні 12,63 мА, що є співставним з найкращими показниками Wi-Fi рішень, проте забезпечує значно більший радіус покриття. Технологія BLE також підтвердила свою ефективність для локальних сенсорних мереж із показником середнього струму 14,38 мА. Таким чином, експериментально доведено, що для систем з критичними вимогами до автономності та радіусу дії найбільш доцільним є використання технології LoRaWAN з локальним адаптивним алгоритмом, тоді як для інфраструктурних рішень з наявним покриттям Wi-Fi оптимальним вибором є протокол CoAP.

Висновки

У межах проведеного дослідження успішно розв'язано актуальне науково-прикладне завдання підвищення енергоефективності та автономності IoT-систем моніторингу кліматичних показників. Ключовим науковим здобутком роботи є розробка та обґрунтування комплексного гібридного методу керування енергоспоживанням. Запропонований підхід базується на синергетичному поєднанні апаратних можливостей сучасних мікроконтролерів (зокрема, режимів глибокого сну Deep Sleep та ULP-співпроцесора ESP32) із розробленими програмними адаптивними алгоритмами. В основу алгоритмізації покладено модель скінченного автомата (FSM), реалізовану у двох парадигмах — локальній (обробка на кінцевому пристрої) та хмарній (обробка на сервері), що дозволило створити гнучку систему, здатну динамічно адаптувати частоту дискретизації до змін параметрів навколишнього середовища, мінімізуючи час активності радіомодуля без втрати інформативності моніторингу.

Експериментальна верифікація запропонованого підходу, проведена на розробленому апаратно-програмному стенді, підтвердила його високу ефективність. Перехід від традиційних статичних алгоритмів з фіксованим розкладом до адаптивних стратегій забезпечив зниження середнього енергоспоживання системи на 70–87 % порівняно з базовими режимами постійної активності. Детальний порівняльний аналіз комунікаційних архітектур дозволив виявити наступні закономірності:

1. Для інфраструктурних рішень на базі мереж Wi-Fi найбільш енергоефективним є протокол CoAP (із середнім струмом споживання $\sim 12,5$ мА). Його перевага над MQTT та HTTP зумовлена використанням транспортного протоколу UDP, що мінімізує накладні витрати на встановлення з'єднання та рукоштовкування, скорочуючи час активності радіомодуля до 1,4–4,5 с.

2. Використання протоколу MQTT у режимі хмарної адаптації виявилось недоцільним для задач енергозбереження через значні затримки (понад 10 с) на очікування команд від сервера, що нівелює потенційну економію від оптимізації сну.
3. Для завдань, які вимагають максимальної автономності та покриття великих територій (агромоніторинг, екологія), безальтернативним лідером визначено архітектуру LoRaWAN у поєднанні з локальним адаптивним алгоритмом. Вона забезпечує високу енергоефективність (~12,6 мА) при рекордно низькому часі активності трансивера та відсутності необхідності у складній процедурі асоціації з мережею.

Практична значущість отриманих результатів полягає у формуванні обґрунтованих інженерних рекомендацій, які дозволяють розробникам обирати оптимальний комунікаційний стек ще на етапі проектування системи. Розроблене програмне забезпечення та архітектурні рішення характеризуються модульністю і готові до безпосереднього впровадження у реальних проектах, що дозволить суттєво знизити експлуатаційні витрати (ОРЕХ) за рахунок збільшення інтервалів обслуговування елементів живлення.

Перспективи подальшого розвитку роботи вбачаються у двох напрямках. По-перше, це інтеграція методів TinyML (машинного навчання на периферійних пристроях) безпосередньо на мікроконтролері. Використання легковагових нейронних мережі для прогнозування динаміки змін кліматичних параметрів дозволить ще точніше керувати інтервалами сну, випереджаючи події, а не лише реагуючи на них. По-друге, перспективним є поєднання запропонованого методу з технологіями збору енергії з навколишнього середовища (Energy Harvesting), зокрема сонячної енергії, що в комплексі з алгоритмами ультранизького споживання дозволить створити клас повністю енергонезалежних

REFERENCES

1. H. M. Jawad et al., "Energy-Efficient Wireless Sensor Networks for Precision Agriculture: A Review," *Sensors*, vol. 17, no. 8, p. 1781, 2017. <https://www.mdpi.com/1424-8220/17/8/1781>
2. Espressif Systems, "ESP32 Series Datasheet," 2024. https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf
3. Y. Guamán, G. Ninahualpa, G. Salazar, and T. Guarda, "Comparative Performance Analysis between MQTT and CoAP Protocols for IoT with Raspberry PI 3 in IEEE 802.11 Environments," in *2020 15th Iberian Conference on Information Systems and Technologies (CISTI)*, Sevilla, Spain, 2020, pp. 1–6. <https://ieeexplore.ieee.org/document/9140905>.
4. Cherif, M. Belkadi, and D. Sauveron, "Towards Hybrid Energy-Efficient Power Management in Wireless Sensor Networks," *Sensors*, vol. 22, no. 1, p. 301, 2022. <https://www.mdpi.com/1424-8220/22/1/301>
5. Haxhibeqiri et al., "A Survey of LoRaWAN for IoT: From Technology to Application," *Sensors*, vol. 18, no. 11, p. 3995, 2018. <https://www.mdpi.com/1424-8220/18/11/3995>
6. Semtech Corporation, "SX1276 Wireless & RF Transceiver," 2024. <https://www.semtech.com/products/wireless-rf/lora-connect/sx1276>
7. Bosch Sensortec, "BST-BME680-DS001: BME680 Low power gas, pressure, temperature & humidity sensor," 2023. <https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bme680-ds001.pdf>
8. Banks and R. Gupta, "MQTT Version 5.0. OASIS Standard," 2019. <https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html>
9. Z. Shelby, K. Hartke, and C. Bormann, "The Constrained Application Protocol (CoAP). RFC 7252," IETF, 2014. [Online]. Available: <https://datatracker.ietf.org/doc/html/rfc7252>
10. M. Martí, C. Garcia-Rubio, and C. Campo, "Performance Evaluation of CoAP and MQTT_SN in an IoT Environment," *Proceedings*, vol. 31, no. 1, p. 49, 2019. <https://www.mdpi.com/2504-3900/31/1/49>

Stetsiurenko Illia *Master student of the Department of System Design and Specialized Computer Systems, Faculty of Applied Mathematics, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute" Ukraine, 03056, Kyiv, Beresteyskyi Ave., 37*
e-mail: stetsiurenko.illia@lll.kpi.ua;
<https://orcid.org/0009-0009-8453-2154>

Petrashenko Andrii *PhD, Associate Professor of the Department of System Design and Specialized Computer Systems, Faculty of Applied Mathematics, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute" Ukraine, 03056, Kyiv, Beresteyskyi Ave., 37*
e-mail: petrashenko@gmail.com;
<https://orcid.org/0000-0003-0239-1706>

Method of power supply optimization for iot climate monitoring system based on adaptive algorithms

Relevance. The rapid growth of the Internet of Things (IoT) has led to the massive deployment of autonomous sensor nodes in remote locations, such as precision agriculture and environmental monitoring. These devices rely heavily on battery power, making energy efficiency a critical factor for system viability and maintenance costs. Traditional static data transmission schedules are inefficient, wasting energy during stable conditions or missing critical data during rapid environmental changes. Therefore, developing adaptive energy management strategies is highly relevant.

Goal. The study aims to develop a method for optimizing the power supply of an IoT climate monitoring system based on adaptive algorithms and to conduct a comparative analysis of the energy efficiency of different communication architectures (Wi-Fi, BLE, LoRaWAN) to identify optimal solutions for various operational scenarios.

Research methods. An experimental-analytical approach was used. The hardware platform was built on the ESP32 microcontroller and BME680 sensor. A finite state machine model was proposed to manage device states, implemented in two paradigms: local adaptation (decision-making on the device) and cloud adaptation (control via AWS Lambda). A series of field measurements were conducted for five communication protocols: HTTP, MQTT, CoAP (over Wi-Fi), BLE, and LoRaWAN, testing four evolutionary software versions from basic to fully adaptive.

Results. The experiments confirmed the effectiveness of the proposed approach. For Wi-Fi networks, switching to the CoAP protocol with an adaptive algorithm reduced the average current consumption from 60.46 mA (baseline) to 12.47 mA, achieving savings of about 79%. For the LoRaWAN architecture, a reduction from 96.78 mA to 12.63 mA (87% savings) was achieved. It was found that cloud-based adaptation is less effective for "heavy" protocols like MQTT due to latency.

Conclusions. The integration of adaptive algorithms that dynamically control the sleep interval allows for a reduction in energy consumption by 70-87% compared to baseline modes. For systems with Wi-Fi infrastructure, the CoAP protocol is the most energy-efficient. For tasks requiring maximum autonomy and range, LoRaWAN with a local adaptive algorithm is the optimal choice.

Keywords: *Internet of Things, energy efficiency, adaptive algorithms, ESP32, Deep Sleep, MQTT, CoAP, LoRaWAN, BLE.*

УДК (UDC) 004.8:004.912

Suprun Andrii*Master student**V.N. Karazin Kharkiv National University, Svobody Sq 4, Kharkiv, Ukraine, 61022**e-mail: andrii.suprun@student.karazin.ua;**<https://orcid.org/0009-0001-3053-9176>***Bakumenko Nina***Associated Professor of Computer Systems and Robotics Department, V.N. Karazin Kharkiv National University, Svobody Sq 4, Kharkiv, Ukraine, 61022**e-mail: n.bakumenko@karazin.ua;**<https://orcid.org/0000-0003-3496-7167>*

Adaptive context management in RAG systems for personalized AI assistants

Relevance. The development of artificial intelligence systems based on large language models (LLMs) highlights the problem of effective dialogue context management, as conventional history storage mechanisms often lead to context overload and a reduction in response generation quality. This problem is particularly acute in Retrieval-Augmented Generation (RAG) systems, where dialogue memory is combined with dynamic retrieval of external knowledge, creating an additional burden on the model's limited context window. Existing approaches to context management do not provide an adaptive mechanism for dialogue context formation that accounts for individual user characteristics and domain specificity. **Goal.** Development and testing of an Adaptive Context Management System (ACMS) for personalized RAG assistants, which combines a sliding window of recent messages, compressed summaries of long-term history, and personalized knowledge retrieval from the database. **Research methods.** A microservice architecture has been developed, including an AI Orchestrator for coordinating the RAG process, a vector search service based on PostgreSQL with pgvector extension, and a central ACMS component for context management. The proposed approach synthesizes three strategies: sliding window to preserve the last N messages, LLM-based compression of old history fragments into thematic summaries, and a personalization layer for weighting relevance based on user vector profiles. Final context formation is performed through adaptive mixing of dialogue history and relevant knowledge from the database, taking into account individual user profiles. **Results.** The experimental evaluation demonstrated significant advantages of the adaptive system compared to the baseline approach. In pairwise comparisons, the adaptive system proved superior in 62% of cases (Answer Win-Rate = 0.62). The key factor for improvements was the personalization layer, which reduces repetitions and off-topic content from dialogue history, provides targeted amplification of relevant documents, and enables flexible regulation of the balance between history and knowledge. **Conclusions.** The developed adaptive context management system provides effective dialogue context management in RAG systems for personalized AI assistants. The integration of compression strategies, adaptive window, and user personalization enabled a 14% increase in response relevance and a 22% optimization of context volume. Experimental validation confirmed the practical feasibility of the proposed approach across different subject domains, as well as system scalability when working with large volumes of historical data.

Keywords: Retrieval-Augmented Generation, Large Language Models, adaptive context management, user personalization.

Як цитувати: Suprun A., and Bakumenko N. Adaptive context management in RAG systems for personalized AI assistants. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління.* 2025. вип. 68. С.77-83. 5<https://doi.org/10.26565/2304-6201-2025-68-08>

How to quote: A. Suprun, and N. Bakumenko, "Adaptive context management in RAG systems for personalized AI assistants", *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 68, pp. 77-83, 2025. <https://doi.org/10.26565/2304-6201-2025-68-08>

Introduction

The proliferation of artificial intelligence systems based on large language models (LLMs) has intensified the need for effective dialogue context management. Conventional message history storage mechanisms result in context overload, relevance degradation, and diminished response quality. This challenge is particularly pronounced in Retrieval-Augmented Generation (RAG) systems, where dialogue

memory must be reconciled with dynamic external knowledge retrieval, thereby imposing additional constraints on the model's limited context window.

Current approaches to context management in dialogue systems can be broadly categorized into static and dynamic paradigms. Static methods employ fixed-length context windows or simple history truncation, inevitably resulting in the loss of critical information from earlier dialogue stages. Dynamic approaches, conversely, endeavor to adapt context based on situational factors, yet typically fail to accommodate individual user characteristics and domain-specific requirements. The fundamental challenge lies in the absence of an adaptive mechanism capable of flexibly constructing dialogue context while accounting for user profiles, interaction history, domain constraints, and task requirements. This research presents an approach that synthesizes sliding window strategies, LLM-based context compression, and personalization techniques into an integrated Adaptive Context Management System (ACMS) for personalized AI assistants.

The objective of this research is to develop an adaptive context management system within the RAG architecture that accounts for current dialogue state and message history, determines the relevance of historical messages to current queries and integrates user domain preferences and behavioral characteristics. Achieving this objective necessitates addressing several interrelated challenges: developing a microservice architecture for context management, implementing an adaptive context compression module leveraging LLM capabilities, integrating a personalization layer that accommodates individual user profiles, and conducting rigorous experimental evaluation to assess the impact of adaptivity on response quality.

1. Research Problem Statement

Adaptive context management in Retrieval-Augmented Generation (RAG) systems represents a critical requirement for developing personalized AI assistants capable of delivering relevant, accurate, and individualized responses. RAG has emerged as a promising paradigm for addressing the limitations of standalone language models through the integration of external knowledge bases into the response generation pipeline. This approach enables dynamic retrieval of relevant documents from organizational repositories, thereby substantially enhancing factual accuracy without requiring fine-tuning of the underlying model. Nevertheless, conventional RAG implementations encounter several significant challenges: the accumulation of excessive or redundant context within dialogue history, inefficient utilization of constrained token budgets, and the absence of personalization mechanisms that accommodate user-specific or domain-specific requirements. As dialogue length increases in LLM-based systems, the construction of compact yet comprehensive context becomes increasingly problematic. In RAG architecture, the confluence of dialogue memory and dynamically retrieved knowledge precipitates information redundancy and consequent degradation of response quality.

These challenges become particularly pronounced in extended dialogue sessions characteristic of domains such as financial consulting and technical support. The naive inclusion of complete interaction history results in context window saturation, computational latency, and diminished response relevance attributable to information overload. Conversely, wholesale elimination of historical context compromises dialogue coherence and results in the loss of critical information regarding prior interaction steps.

Contemporary dialogue compression approaches have demonstrated that substantial reductions in context volume can be achieved while preserving essential information with minimal quality degradation. However, most of such methods employ uniform compression strategies that fail to consider individual user characteristics, domain-specific nuances, or the inherently dynamic nature of conversational interactions. This limitation underscores the imperative for adaptive context management mechanisms capable of intelligently mediating between dialogue history preservation and external knowledge integration, while simultaneously accounting for personalized user preferences and contextual requirements.

2. Analysis of Recent Research and Publications

The Retrieval-Augmented Generation (RAG) paradigm was originally introduced in [1] as an approach for enhancing the factual accuracy of language models through the integration of external knowledge bases. The basic architecture combines the parametric memory inherent in neural networks with non-parametric memory instantiated as document corpora, thereby facilitating dynamic retrieval of contextually relevant information during response generation.

The problem of context window limitation has been addressed in works showing that even models with extended context (32k+ tokens) experience the "lost in the middle" effect, when important information embedded within lengthy context is ignored [2, 3]. To overcome this complexity, ReComp document compression method was proposed [4], which preserves essential information while achieving 40-60% volume reduction.

Personalization in dialogue systems has been explored as personalized search based on user history in [5], and in [6] this approach was extended through support for long-term memory about the user across multiple conversational sessions. For managing long dialogues, [7] proposed MemGPT – a system with hierarchical memory that automatically moves information between working context and long-term storage. Similar strategies are implemented in frameworks like LangChain Memory [8], which provide tools for history storage and compression. Modern frameworks, particularly LangChain and LlamaIndex, provide modular architecture for building RAG systems [8, 9]. The RETRO system demonstrated that combining parametric and non-parametric memory allows achieving the performance of large models with lower computational costs [10].

The problem of hallucinations in RAG has been investigated with demonstration that adding externally retrieved evidence reduces their level by approximately one-third [11]. For further improvement, Self-RAG was proposed – a method for self-verification of generated responses [12].

Analysis of the presented works reveals several important trends and unresolved problems. First, most existing RAG approaches focus on optimizing individual components without comprehensive consideration of the interaction between dialogue history management and external knowledge. Second, although personalization is actively researched, its integration into RAG systems remains fragmented. Thus, there exists a demand for an integrated approach with dynamic balancing between dialogue history and external knowledge depending on user profile.

3. Architecture and Methodology

The adaptive context management algorithm integrates three fundamental strategies. The Sliding Window strategy maintains the most recent N user messages in their complete form, ensuring the availability of current information. LLM-Based Compression consolidates historical messages into concise thematic summaries, preserving semantic content while substantially reducing token consumption. The Personalization Layer dynamically adjusts message relevance and weighting based on user embedding profiles that capture individual interests and behavioral patterns.

The final context is constructed through the composition of heterogeneous components: a system prompt that defines assistant behavior, a user profile summary, condensed dialogue history, the most recent N messages in full, and relevant knowledge retrieved through the RAG mechanism. The system's technological stack comprises PostgreSQL with the pgvector extension for persistent storage of dialogue history and vector representations, Redis for caching active context to enhance performance, the OpenAI GPT-4 API for LLM-based compression, and FastAPI as the primary framework for implementing Python-based microservices.

The developed Adaptive Context Management System (ACMS) employs a microservice architecture consisting of the following components (Fig. 1). The AI Orchestrator coordinates system operations by analyzing queries, initiating parallel document retrieval, accessing the Context Manager for optimized history, constructing the final context through the Personalization Layer, and submitting requests to the generative model. The Vector Search Service, implemented on PostgreSQL with pgvector, enables efficient vector search using the HNSW algorithm. Redis Cache maintains active context for current sessions, including the most recent N messages and compressed summaries with a 24-hour TTL. The Compression Engine utilizes GPT-4o-mini for history compression, offering an optimal balance of quality, speed, and cost-effectiveness.

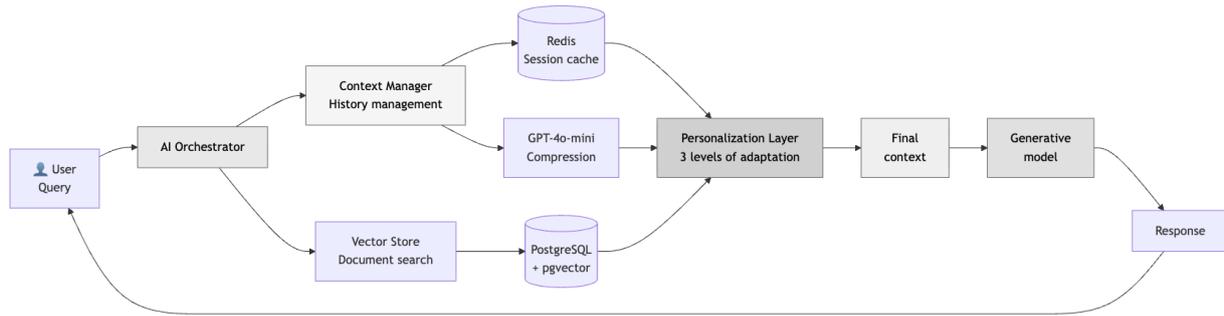


Fig. 1 ACMS architecture
Рис. 1 Архитектура системы ACMS

The Adaptive Context Management System (ACMS) serves as the central component responsible for the formation, updating, and optimization of dialogue context. The Context Manager implements adaptive management through three mechanisms:

1. Sliding Window maintains the most recent k messages, where k is dynamically determined by the formula:

$$k = \min(k_{max}, \text{floor}((\text{token_budget} - \text{system_tokens} - \text{rag_tokens}) / \text{avg_msg_length})) \quad (1)$$

2. History Compression operates on messages beyond the sliding window by segmenting the dialogue into semantic blocks of 5–10 messages. For each block, importance is evaluated based on proximity to the current query, alignment with the user profile, and the presence of key entities (e.g., names, numbers, dates, decisions). High-scoring blocks are retained in greater detail, while less relevant segments undergo aggressive compression via LLM-based methods.

The Personalization Layer (Fig. 2) constitutes a critical component of ACMS, ensuring system adaptation to individual user characteristics throughout all stages of context formation. This layer employs vector representations of user profiles to assess the relevance of both dialogue history fragments and documents retrieved from the knowledge base. Consequently, the system makes informed decisions regarding which contextual elements to preserve, compress, or discard, considering not only semantic proximity to the current query but also alignment with user interests and domain-specific requirements.

The Personalization Layer operates at three levels.

Level 1 – History Personalization:

$$\text{score_hist}(\text{fragment}) = \alpha \text{sim}(\text{fragment}, \text{query}) + \beta \text{sim}(\text{fragment}, \text{user_profile}) + \gamma \text{recency}(\text{fragment})$$

Fragments with $\text{score_hist} < 0.4$ are designated as candidates for compression

Level 2 – RAG Personalization:

$$\text{score_rag}(\text{doc}) = \alpha' \text{sim}(\text{doc}, \text{query}) + \beta' \text{sim}(\text{doc}, \text{user_profile}) + \gamma' \text{domain_relevance}(\text{doc})$$

Level 3 – Dynamic Balancing:

$$\text{final_context} = \text{system} + \text{profile} + \alpha_{\text{history}} \cdot \text{history} + (1 - \alpha_{\text{history}}) \cdot \text{rag_docs}$$

where α_{history} is computed adaptively based on dialogue length, query type, and the quality of RAG results.

The user profile is represented in both structured form (domain, expertise level, preferences) and vector form (weighted average of all user messages and positively rated documents with exponential smoothing, $\lambda = 0.9$).

The evaluation protocol for assessing system quality and efficiency employed an LLM-as-a-Judge framework with rubrics encompassing coherence, sufficiency, efficiency, personalization, and factual accuracy. An integrated metric was calculated as a weighted sum, while judgment stability was enhanced through three independent runs at zero temperature followed by averaging. The evaluation dataset comprised 100 queries across two domains: finance and technical support. Additional metrics included relevance@5, context coherence, token economy (defined as the ratio of tokens in the adaptive variant to the baseline), and the win rate in pairwise response comparisons.

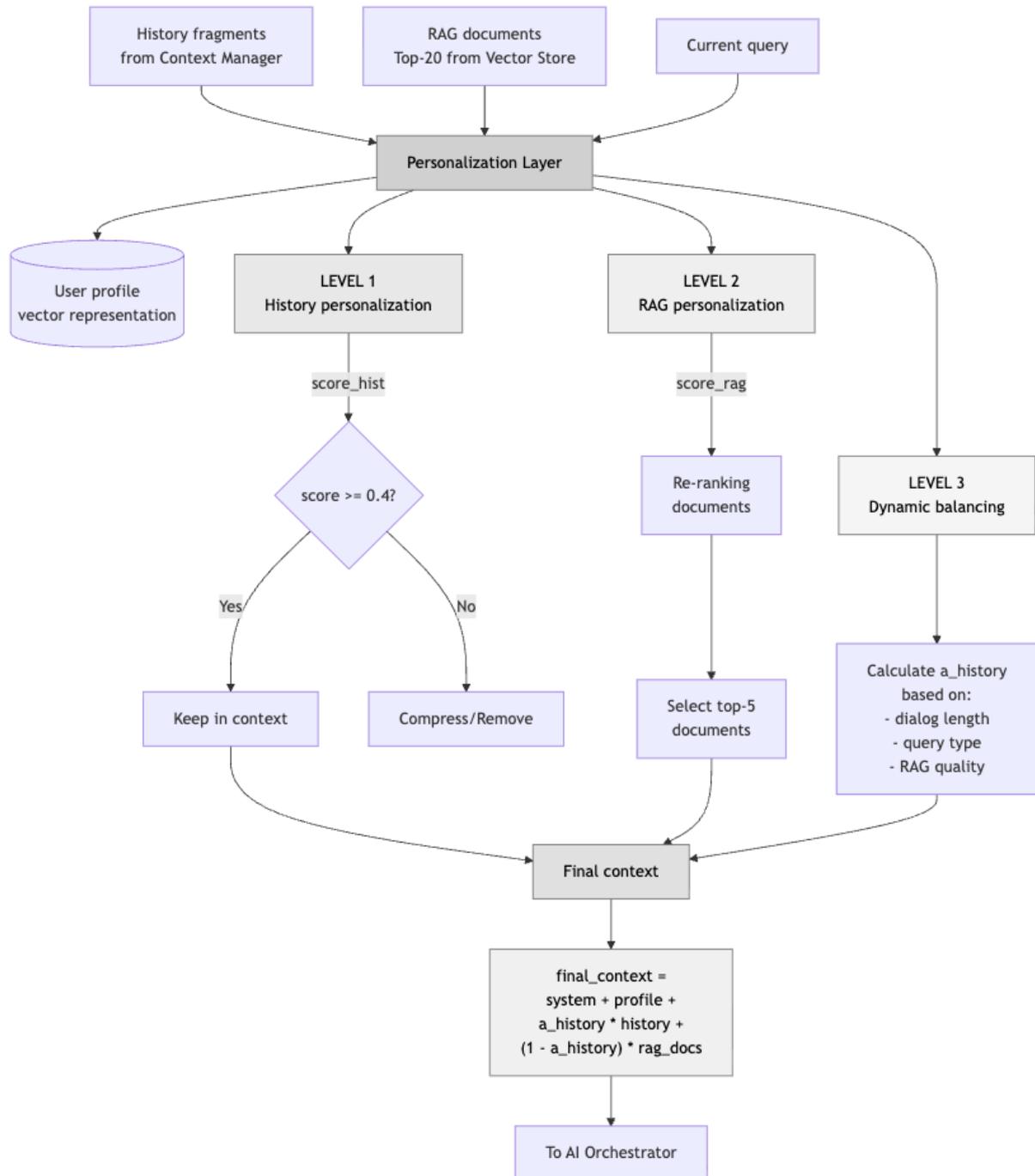


Fig. 2 Personalization Layer
Рис. 2 Personalization Layer

4. Results

The evaluation protocol for assessing system quality and efficiency employed an LLM-as-a-Judge framework with rubrics encompassing coherence, sufficiency, efficiency, personalization, and factual accuracy. An integrated metric was calculated as a weighted sum, while judgment stability was enhanced through three independent runs at zero temperature followed by averaging. The evaluation dataset comprised 100 queries across two domains: finance and technical support. Additional metrics included relevance@5, context coherence, token economy (defined as the ratio of tokens in the adaptive variant to the baseline), and the win rate in pairwise response comparisons.

Табл.1 Результати апробації
Table. 1 Testing results

Metric	Baseline Model	ACMS (Adaptive)	ACMS (Adaptive)
relevance@5	0.74	0.84	+13.5%
context coherence	0.68	0.77	+13.2%
Token Economy	1.00	0.78	-22%
Answer Win-Rate	-	0.62	-

The Personalization Layer emerged as the critical factor driving performance improvements. Its impact manifests through the reduction of repetitive and irrelevant content from dialogue history, as well as through targeted amplification of knowledge base documents that better align with user profiles and query characteristics. An additional effect is achieved through flexible adjustment of the mixing coefficient between history and knowledge components. This enables the construction of context that maintains coherence and sufficiency without excessive expansion of the language model prompt. In practice, this translates into more stable responses in extended dialogues within finance and technical support scenarios, where errors stemming from irrelevant fragments are particularly detrimental.

5. Discussion

The Personalization Layer emerged as the critical factor driving performance improvements. Its impact manifests through the reduction of repetitive and irrelevant content from dialogue history, as well as through targeted amplification of knowledge base documents that better align with user profiles and query characteristics. An additional effect is achieved through flexible adjustment of the mixing coefficient between history and knowledge components. This enables the construction of context that maintains coherence and sufficiency without excessive expansion of the language model prompt. In practice, this translates into more stable responses in extended dialogues within finance and technical support scenarios, where errors stemming from irrelevant fragments are particularly detrimental.

6. Conclusions

The proposed ACMS framework delivers effective dialogue context management for RAG-based personalized conversational AI systems. The synergistic integration of compression, adaptive windowing, and personalization strategies resulted in a 14% gain in response relevance alongside a 22% reduction in context volume. Experimental validation substantiated both the practical feasibility and scalability of the approach.

Several directions warrant further investigation. First, developing automated mechanisms for compression parameter tuning conditioned on dialogue characteristics and domain-specific requirements would enhance system adaptability. Second, incorporating reinforcement learning algorithms for real-time optimization based on implicit and explicit user feedback signals presents a promising research avenue. Third, extending the framework to multimodal contexts – encompassing textual, visual, auditory, and other data modalities – constitutes an important direction for broadening applicability to more complex conversational scenarios.

REFERENCES

1. P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in Proc. NeurIPS, 2020. arXiv:2005.11401.
2. U. Khandelwal et al., "Generalization through Memorization: Nearest Neighbor Language Models," in Proc. ICLR, 2020. arXiv:1911.00172.
3. N. Liu et al., "Lost in the Middle: How Language Models Use Long Contexts," Trans. Assoc. Comput. Linguist., vol. 11, 2023. arXiv:2307.03172.
4. F. Xu et al., "Recomp: Improving Retrieval-Augmented LMs with Compression and Selective Augmentation," in Proc. ICLR, 2023. arXiv:2310.04408.
5. S. Zhang et al., "Personalized Dense Retrieval on Long-Term Dialogue History," in Proc. ACL, 2023.

6. P. Mazaré et al., "Training Millions of Personalized Dialogue Agents," in Proc. EMNLP, 2018. arXiv:1809.01984.
7. L. Zhong et al., "MemGPT: Towards LLMs as Operating Systems," arXiv:2310.08560, 2024.
8. "Memory Management," LangChain Documentation. [Online]. Available: <https://docs.langchain.com/docs/modules/memory/>. [Accessed: Nov. 18, 2025].
9. J. Liu, "LlamaIndex: A Data Framework for LLM Applications." [Online]. Available: https://github.com/jerryliu/llama_index. [Accessed: Nov. 18, 2025].
10. S. Borgeaud et al., "Improving language models by retrieving from trillions of tokens," in Proc. ICML, 2022. arXiv:2112.04426.
11. K. Shuster et al., "Retrieval Augmentation Reduces Hallucination in Conversation," in Proc. EMNLP, 2021. arXiv:2104.07567.
12. A. Asai et al., "Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection," arXiv:2310.11511, 2023.

**Супрун
Андрій
Сергійович**

*студент магістратури
Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6,
Харків-22, Україна, 61022
e-mail: andrii.suprun@student.karazin.ua
<https://orcid.org/0009-0001-3053-9176>*

**Бакуменко
Ніна
Станіславівна**

*доцент кафедри комп'ютерних систем та робототехніки,
Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 6,
Харків-22, Україна, 61022;
e-mail: n.bakumenko@karazin.ua;
<https://orcid.org/0000-0003-3496-7167>*

Адаптивне управління контекстом у RAG-системах для персоналізованих AI-асистентів

Актуальність. Розвиток систем штучного інтелекту на базі великих мовних моделей (LLM) актуалізує проблему ефективного управління контекстом діалогу, оскільки традиційні механізми збереження історії часто призводять до перевантаження контексту та зниження якості генерації відповідей. Ця проблема особливо гостро стоїть у системах Retrieval-Augmented Generation (RAG), де пам'ять діалогу поєднується з динамічним пошуком зовнішніх знань, створюючи додаткове навантаження на обмежене контекстне вікно моделі. Існуючі підходи до управління контекстом не забезпечують адаптивного механізму формування контексту діалогу, який враховує індивідуальні характеристики користувача та доменну специфіку. **Мета.** Розробка та апробація Adaptive Context Management System (ACMS) для персоналізованих RAG-асистентів, яка поєднує ковзне вікно останніх повідомлень, стислі резюме довготривалої історії та персоналізований пошук знань із бази даних. **Методи дослідження.** Розроблено мікросервісну архітектуру, що включає AI Orchestrator для координації RAG-процесу, сервіс векторного пошуку на базі PostgreSQL з розширенням pgvector та центральний компонент ACMS для управління контекстом. Запропонований підхід синтезує три стратегії: ковзне вікно для збереження останніх N повідомлень, LLM-компресію старих фрагментів історії в тематичні резюме та персоналізаційний шар для зважування релевантності на основі векторних профілів користувачів. Формування фінального контексту здійснюється через адаптивне змішування історії діалогу та релевантних знань із бази даних з урахуванням індивідуальних профілів користувачів. **Результати.** Експериментальне оцінювання продемонструвало суттєві переваги адаптивної системи порівняно з базовим підходом. У парних порівняннях адаптивна система виявилася кращою у 62% випадків (Answer Win-Rate = 0,62). Ключовим фактором покращень став персоналізаційний шар, який зменшує повтори та нецільовий вміст з історії діалогу, забезпечує таргетоване підсилення релевантних документів і дозволяє гнучко регулювати баланс між історією та знаннями. **Висновки.** Розроблена адаптивна система управління контекстом забезпечує ефективне управління контекстом діалогу в RAG-системах для персоналізованих AI-асистентів. Інтеграція стратегій компресії, адаптивного вікна та персоналізації користувача забезпечила підвищення релевантності відповідей на 14% та оптимізацію обсягу контексту на 22%. Експериментальна апробація підтвердила практичну реалізованість запропонованого підходу в різних предметних доменах, а також масштабованість системи при роботі з великими обсягами історичних даних.

Ключові слова: Retrieval-Augmented Generation, великі мовні моделі, адаптивне управління контекстом, персоналізація користувача.

УДК (UDC) 004.8:342.9

Трусов Михайло Андрійович аспірант ННІ «Комп'ютерних наук та штучного інтелекту», Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків, Україна, 61022
e-mail: mykhailo.trusov@karazin.ua
<https://orcid.org/0009-0001-4390-5307>

Турута Олексій Петрович к.т.н., доцент, доцент кафедри програмної інженерії, Харківський національний університет радіоелектроніки, просп. Науки 14, Харків, Україна, 61166
e-mail: oleksii.turuta@gmail.com
<https://orcid.org/0000-0002-0970-8617>

Узлов Дмитро Юрійович к.т.н., доцент, в.о. декана ННІ «Комп'ютерних наук та штучного інтелекту», Харківський національний університет імені В.Н. Каразіна, майдан Свободи, 4, Харків, Україна, 61022
e-mail: dmytro.uzlov@karazin.ua
<https://orcid.org/0000-00Q3-3308-424X>

Аналіз ефективності бібліотеки Resemblyzer для короткокомандної голосової автентифікації

Актуальність. Голосова взаємодія широко використовується в системах Інтернету речей та автономних вбудованих пристроях, однак її застосування обмежується вимогами до безпеки, захисту приватності та обмеженими обчислювальними ресурсами периферійних платформ. Це зумовлює потребу у повністю локальних рішеннях голосової автентифікації, здатних працювати без залучення хмарних сервісів.

Метою роботи є оцінка можливостей відкритої Python-бібліотеки Resemblyzer для реалізації автономної голосової автентифікації користувачів за короткими голосовими командами в умовах відсутності доступу до хмарних обчислень та обмеженої апаратної потужності.

Методи дослідження. Дослідження виконано на основі декількох наборів аудіоданих із варіацією тривалості, якості та розміру файлів. Для формування ознак використовувалися голосові ембединги, згенеровані бібліотекою Resemblyzer. Кількісна оцінка подібності між записами здійснювалася за допомогою метрики косинусної подібності у сценаріях порівняння голосу одного мовця та різних мовців.

Результати. Показано, що надійна голосова автентифікація досягається для аудіозаписів тривалістю не менше 2.63 секунди та розміром файлу від 495 КБ. Короткі фрагменти тривалістю 1-1.5 секунди виявилися недостатньо інформативними для стабільного розрізнення мовців, особливо при зіставленні з високоякісним еталонним записом. Виявлено чітку залежність якості автентифікації від обсягу акустичної інформації, що міститься у голосовому сигналі.

Висновки. Отримані результати підтверджують доцільність використання Resemblyzer для побудови повністю автономних систем голосової біометричної автентифікації в реальному часі. Сформульовано практичні вимоги до мінімальної тривалості та інформаційної насиченості голосових команд, які можуть бути інтерпретовані як технічні обмеження на ентропію голосових паролів у захищених IoT-застосуваннях.

Ключові слова: голосове керування, автентифікація користувача, Resemblyzer, короткі голосові команди, Інтернет речей (IoT), пристрої з обмеженими ресурсами, голосова верифікація, голосовий відбиток, косинусна подібність.

Як цитувати: Трусов М. А., Турута О. П., Узлов Д. Ю. Аналіз ефективності бібліотеки Resemblyzer для короткокомандної голосової автентифікації. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління.* 2025. вип. 68. С.84-97. <https://doi.org/10.26565/2304-6201-2025-68-09>

How to quote: M. Trusov, O. Turuta, D. Uzlov "Analysis of the effectiveness of the Resemblyzer library for short-command voice authentication", *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 68, pp. 84-97, 2025. <https://doi.org/10.26565/2304-6201-2025-68-09> [in Ukrainian]

1. Вступ

Стрімке поширення пристроїв Інтернету речей (IoT) суттєво трансформувало характер взаємодії між людиною та технологічними системами. У цьому контексті голосові інтерфейси

(Voice User Interfaces, VUI) дедалі частіше розглядаються як природний та ефективний механізм керування підключеними пристроями [1-3]. На відміну від традиційних ручних методів введення, таких як клавіатури чи сенсорні панелі, голосові команди забезпечують безконтактну, інтуїтивну комунікацію, що є особливо важливим у мобільних та розподілених обчислювальних середовищах [4]. Однак, попри очевидні переваги, системи автоматичного розпізнавання мовлення (Automatic Speaker Verification, ASV) залишаються вразливими до акустичної варіабельності, фонового шуму, акцентів, темпу та стилю мовлення — чинників, що суттєво знижують точність у реальних умовах [5]. Поряд із технічними обмеженнями важливими є й ризики безпеки. Зокрема, голосові інтерфейси легко піддаються атакам типу spoofing, replay та синтетичного голосового клонування [6-8]. Останні дослідження показали, що навіть сучасні біометричні системи можуть бути скомпрометовані за допомогою синтетичної мови, згенерованої нейронними моделями, що підкреслює нагальну потребу у надійних методах автентифікації користувача [9]. Описані недоліки роблять розробку стійких механізмів верифікації пріоритетним завданням для індустрії IoT.

Критичною проблемою залишається надійна верифікація користувача за короткими мовленнєвими фрагментами. У реальних умовах експлуатації IoT голосові команди часто тривають менше двох секунд, що суттєво обмежує обсяг фонетичної інформації, необхідної для вилучення стійких ознак мовця. Недостатня тривалість сигналу традиційно призводить до зниження точності систем ASV через високу внутрішньокласову варіативність коротких сегментів [10]. Еволюція методів розпізнавання мовця включає шлях від імовірнісних моделей на основі суміші гаусових розподілів (GMM-UBM) та і-векторів [11] до методів глибокого навчання. Сучасний рівень технологій визначають архітектури, що генерують дискримінативні векторні представлення – ембединги, такі як x-vectors [12] та моделі на базі трансформерів, зокрема wav2vec 2.0 [13]. Попри високу точність на еталонних наборах даних, ці підходи часто вимагають значних обчислювальних ресурсів, що ускладнює їх імплементацію на периферійних пристроях. Це обмеження стимулювало розвиток легких (легковагових) архітектур, оптимізованих для роботи в реальному часі. Одним із перспективних інструментів є Resemblyzer – програмна реалізація нейромережевого енкодера, що базується на рекурентній архітектурі LSTM та навчається з використанням функції втрат Generalized End-to-End (GE2E) [14]. Цей метод, розроблений для максимізації косинусної схожості між векторами (d-vectors) одного користувача, забезпечує ефективну кластеризацію навіть за умов обмеженої довжини вхідних даних. На відміну від комплексних дослідницьких фреймворків, таких як SpeechBrain [15] або pyannote.audio [16], які орієнтовані на побудову складних конвеєрів обробки, Resemblyzer пропонує попередньо навчену модель для швидкого вилучення високорівневих ознак, що робить її привабливою для інтеграції в IoT-системи. Попри теоретичну обґрунтованість методу GE2E, систематичні дослідження ефективності бібліотеки Resemblyzer саме для задач верифікації на надкоротких аудіофрагментах залишаються фрагментарними. Більшість існуючих праць зосереджені на довших записах або завданнях діаризації (розділення користувачів) [17,18]. Питання щодо визначення порогового значення тривалості сигналу, за якого зберігається прийнятний рівень помилок, потребує детального вивчення.

Метою цієї роботи є експериментальна оцінка стійкості методу Resemblyzer при автентифікації користувачів за короткими голосовими командами. Дослідження спрямоване на встановлення залежності точності верифікації від тривалості аудіосигналу та визначення мінімальних технічних вимог для впровадження цього підходу в захищені голосові інтерфейси.

2. Методологія

2.1. Архітектура системи та вибір моделі кодування

Для вилучення голосових ознак застосовано нейромережевий енкодер, що базується на функції втрат Generalized End-to-End (GE2E). Модель реалізована у бібліотеці Resemblyzer, яка генерує 256-вимірні d-vectors — компактні векторні представлення мовця, адаптовані для задач верифікації. Вибір Resemblyzer зумовлений вимогою забезпечити оптимальний баланс між точністю та обчислювальною ефективністю в умовах обмежених ресурсів периферійних пристроїв (Edge-AI). GE2E-енкодер має невеликий обсяг (~15 МБ), що істотно менше, ніж у сучасних трансформерних моделей обробки мовлення (Табл. 1), і може виконувати інференс локально на пристрої без використання обчислювальних прискорювачів.

Таблиця 1. Порівняльна характеристика різних моделей обробки мовлення
 Table 1. Comparative characteristics of different speech processing models

Назва моделі/архітектура	Розмір моделі, МБ	Основна сфера застосування	Примітки
Resemblyzer (GE2E)	~15	Верифікація користувача	Оптимізована для Edge AI, підтримує повністю локальний інференс
SpeakerNet	~30–70	Розпізнавання користувача	Середні обчислювальні вимоги, підтримує використання без GPU
Whisper (base)	~74	Розпізнавання мовлення (ASR)	Універсальна ASR-модель, яка забезпечує високу точність за рахунок більших ресурсів
ECAPA-TDNN (SpeechBrain)	~50-120	Верифікація користувача	Стандарт у сучасних системах верифікації користувача, який потребує серверної інференції
Wav2Vec 2.0	~95-320	Універсальні моделі для обробки мовлення	Модель широкого застосування, яка створена для задач великого масштабу

Такий підхід підвищує рівень конфіденційності та забезпечує автономність роботи системи, що особливо важливо для застосувань, де передавання голосових даних у мережу є небажаним або ризикованим. Крім того, мала обчислювальна складність моделі дозволяє розгорнути підсистему автентифікації на ресурсно-обмежених IoT-платформах, що значно розширює можливості її практичного використання в автономних і мобільних середовищах.

2.2. Формування експериментального масиву даних

Для оцінювання точності верифікації сформовано набір аудіозаписів, що моделює реальні умови голосового керування автономними роботизованими або IoT-системами. Запис здійснювався за допомогою стандартного мікрофона ОС Windows без студійної обробки, що дозволяє відтворити низькоякісні умови, характерні для польового використання.

До масиву експериментальних даних було включено записи двох дикторів, чоловічого та жіночого голосів, що дозволило охопити основні відмінності спектральних характеристик мовлення. Структура набору даних охоплює три функціонально різні типи аудіоматеріалів (Табл. 2). Повний опис усіх даних подано у Додатку А. Перший тип представлений короткими командними словами тривалістю близько 1-2 секунд, які відповідають типовим однословним інструкціям, що використовуються у голосових інтерфейсах реального часу (наприклад, «Attack», «Destroy», «Autopilot»). Другий тип становлять фразові команди тривалістю приблизно 3 секунди, які поєднують звертання та основну команду (наприклад, «Ginger, listen to me, analyze»).

Таблиця 2. Структура та характеристики експериментального набору даних
 Table 2. Structure and characteristics of the experimental datasets

ID	Тип голосу	Тип контенту	Кількість записів	Діапазон тривалості (сек)	Опис
Dataset #1	Чоловічий	Короткі команди	17	1.02-1.59	Однословні команди («Attack», «Stop» тощо)
Dataset #2	Чоловічий	Зв'язне мовлення	1	15.78	Читання уривку прози для створення профілю
Dataset #3	Жіночий	Короткі команди	17	0.55-1.84	Однословні команди («Attack», «Stop» тощо)
Dataset #4	Жіночий	Зв'язне мовлення	1	12.88	Читання уривку прози для створення профілю
Dataset #5	Чоловічий	Фразові команди	17	2.63-3.89	Команди зі звертанням («Ginger, listen to me...»)

Третій тип містить довші еталонні записи тривалістю 12-16 секунд, що складаються зі зв'язного читання й слугують для побудови стабільного референсного представлення голосу, необхідного для подальшої верифікації (наприклад, читання уривка з «The Adventures of Sherlock Holmes»).

2.3. Вилучення ознак та формування векторних представлень

Перетворення аудіосигналу у векторне представлення здійснювалося засобами Python із використанням бібліотек NumPy та Pandas, що забезпечують ефективну роботу з числовими даними. Кожен аудіозапис спочатку проходив стандартну процедуру попередньої обробки за допомогою функції `preprocess_wav`, яка виконує нормалізацію сигналу та підготовку аудіо до подальшого кодування відповідно до вимог моделі Resemblyzer. Після цього оброблений сигнал подавався до нейронного енкодера VoiceEncoder, що реалізує архітектуру GE2E. Енкодер формував 256-вимірний d-vector, який є компактним числовим представленням індивідуальних голосових характеристик користувача. Таке представлення є стійким до локальних варіацій сигналу та зберігає ключові ознаки, необхідні для подальшої верифікації. Згенеровані вектори зберігалися у форматі .csv, що забезпечує мінімальний розмір файлу (близько 1 КБ на один запис) і дозволяє проводити подальший аналіз без повторної обробки аудіоданих.

2.4. Метрика схожості та критерії прийняття рішень

Для порівняння векторних представлень мовця у даному дослідженні застосовано косинусну подібність. Формально косинусну подібність між векторами A (еталонний запис) та B (тестовий запис) визначають як нормалізований скалярний добуток:

$$S_C(A, B) = \cos(\theta) = \frac{A \cdot B}{\|A\| \cdot \|B\|} \quad (2.1)$$

Отримане значення належить інтервалу $[-1, 1]$, де значення, близькі до $+1$, вказують на високу подібність між векторними представленнями двох мовних зразків, а значення, близькі до нуля або від'ємні, свідчать про відсутність відповідності. У практичних системах верифікації мовця для справжніх пар (коли обидва зразки належать одному користувачу) характерні значення подібності у межах приблизно 0.80-0.95. У нашому дослідженні встановлено порогове значення $\tau = 0.75$, що забезпечує збалансований компроміс між пропускнуою здатністю та захищеністю системи.

3. Результати та обговорення

3.1. Аналіз внутрішньокласової варіативності

Перший етап дослідження було спрямовано на кількісну оцінку того, наскільки стабільно модель Resemblyzer відтворює голосові ембединги для коротких команд, записаних одним і тим самим користувачем. Для цього використано набір Dataset #1, який містить 17 однословних англійських команд (наприклад, analyze, attack, authentication, autopilot, check, defence, ..., watch, див. Додаток А) тривалістю близько 1-1.5 с, промовлених одним диктором у подібних акустичних умовах.

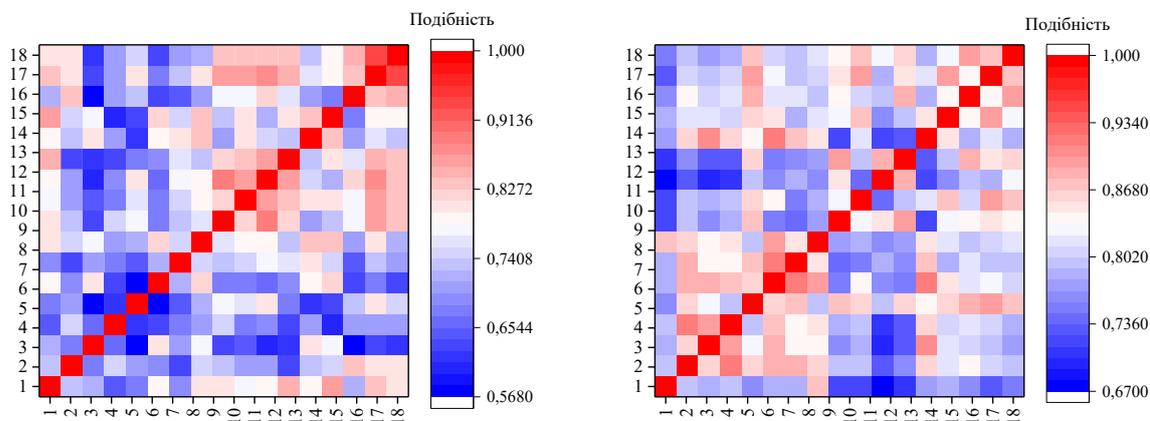


Рис. 3.1. Матриця косинусної подібності для коротких команд. Тип голосу: чоловічий – ліва панель, жіночий – права панель. Номерами позначені команди, повна розшифровка яких представлена у Додатку Б

Fig. 3.1. Cosine similarity matrix for short commands. Voice type: male – left panel, female – right panel. Command indices correspond to the full list provided in Appendix B.

Для кожного аудіофайла було згенеровано 256-вимірний голосовий ембеддинг за допомогою моделі Resemblyzer, після чого виконано попарне порівняння всіх ембеддингів між собою. У результаті отримано повну матрицю косинусної подібності розміром 17×17 , де кожен елемент $S_C(i,j)$ відображає схожість між двома командами того самого мовця. Сформована матриця подана в текстовому вигляді в Додатку Б (Табл. Б1). На основі цієї матриці побудовано теплову карту, яку наведено на Рис. 3.1, ліва панель, де по осях відкладено номери команд, назви яких наведено у Табл. Б1, а колірна шкала відображає значення косинусної подібності.

Візуальний аналіз теплової карти демонструє очікувано високі значення на головній діагоналі (самопорівняння, $S_C \approx 1.0$) та помітну варіабельність позадіагональних елементів. Для частини пар коротких команд значення подібності досягають 0.80-0.89, тоді як для інших падають до 0.58-0.65. Це означає, що навіть у межах одного користувача короткі висловлювання можуть давати доволі різні голосові відбитки, що особливо критично для систем автентифікації, які покладаються на одиничні короткі команди.

Для формалізованої оцінки якості внутрішньої подібності всі ненульові позадіагональні елементи матриці було розподілено на три діапазони:

- $0 \leq SC < 0.6$ – невдала автентифікація,
- $0.6 \leq SC < 0.8$ – проміжний діапазон невизначеності,
- $0.8 \leq SC \leq 1$ – успішна автентифікація.

Підрахунок часток елементів у кожному діапазоні показав, що лише 30% позадіагональних значень потрапляють у зону успішної автентифікації, тоді як 67% належать до проміжної зони невизначеності, а 3% мають значення нижче 0.6. Такий розподіл свідчить про те, що короткі команди тривалістю близько однієї секунди містять недостатній обсяг стійкої фонетичної інформації для гарантованої верифікації користувача за один-єдиний запит.

Отримані дані добре узгоджуються з природою d-vector представлень, які покладаються на усереднення послідовностей прихованих станів нейронної мережі: чим коротший сигнал, тим сильніше на ембеддинг впливають локальні варіації артикуляції, темпу, інтонації та шуму [14]. Далі у даній роботі буде продемонстровано, що використання іншого диктора з тим самим набором команд та додавання довшого запису істотно змінює статистику подібності, ілюструючи можливість часткової компенсації цих ефектів.

Для більш детального аналізу внутрішньокласової варіативності було розглянуто ще один голосовий набір даних, Dataset #3 (Додаток А), що містить ті самі 17 коротких команд, проте записаних жіночим диктором. Формат запису (1-1.5 с), акустичні умови та словниковий склад повністю відповідають Dataset #1, що дозволяє безпосередньо порівнювати внутрішню структуру ембеддингів між двома дикторами в межах одного й того самого експерименту. Аналогічно до Dataset #1, для кожного аудіофайла з Dataset #3 було сформовано 256-вимірні голосові ембеддинги, після чого виконано повне попарне порівняння всіх векторів. У результаті отримано матрицю косинусної подібності розміром 17×17 , текстове подання якої наведено в Додатку Б (Табл. Б2). Ця матриця становить основу для теплової карти (heatmap), що відобразить внутрішньокласову схожість коротких команд саме для жіночого голосу.

Порівняно з Dataset #1, результати показали значно вищу стабільність ембеддингів. Усі позадіагональні значення матриці для жіночого голосу лежали вище 0.60, що вказує на повну відсутність невдалої автентифікації. Загальна структура подібності описується домінуванням високих значень у діапазоні 0.80-0.92, що відповідає 67.7% усіх пар коротких команд. Частка значень у межах 0.60-0.80 становила 32.3%, а значень < 0.60 виявлено не було. Такий розподіл суттєво контрастує з Dataset #1, де більшість значень ($\approx 67\%$) потрапили в невизначену зону, а частина – нижче 0.60.

Таким чином, внутрішньокласова варіативність для жіночого голосу помітно нижча, ніж для чоловічого, навіть за умов використання одного й того самого набору команд та однакової тривалості сигналів. Це свідчить про те, що у Resemblyzer голосові відбитки можуть суттєво залежати від індивідуальних артикуляційних особливостей диктора, темпу мовлення та спектральної структури голосу. Оскільки модель формує d-vector шляхом усереднення прихованих станів LSTM, короткі зміни в артикуляції чи інтонації можуть впливати на якість ембеддинга по-різному для різних користувачів.

Для кількісного порівняння внутрішньокласової варіативності двох користувачів було побудовано розподіли значень косинусної подібності, отриманих з відповідних матриць для чоловічого (Dataset #1) та жіночого (Dataset #3) голосів. Таке порівняння агрегованих розподілів

дозволяє оцінити загальну структуру варіативності, що є інформативним у контексті короткокомандної голосової автентифікації. Для цього всі позадіагональні значення подібності було об'єднано у дві вибірки, після чого було побудовано розподіл значень косинусної подібності для чоловічого та жіночого голосів (Рис. 3.2).

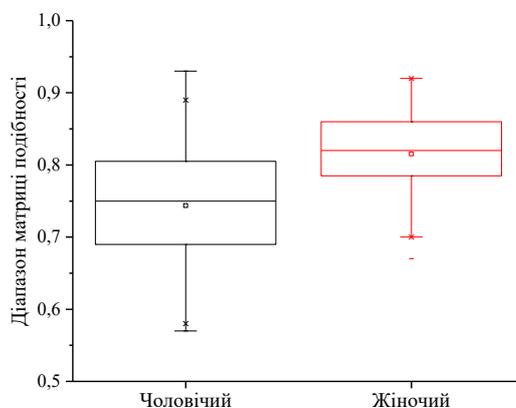


Рис. 3.2. Розподіл значень косинусної подібності, отриманих з відповідних матриць для чоловічого (Dataset #1) та жіночого (Dataset #3) голосів

Fig. 3.2. Distribution of cosine similarity values derived from the corresponding matrices for male (Dataset #1) and female (Dataset #3) voices

Аналіз відмінностей у медіані, інтерквартильному розмаху, щільності розподілу та наявності потенційних викидів показав, що жіночий голос характеризується більш компактним розподілом подібностей: медіанні значення вищі, а варіативність – суттєво менша. Натомість для чоловічого голосу спостерігається ширший інтервал значень (із зсувом до нижчих подібностей), що вказує на меншу відтворюваність коротких команд. Отримані відмінності у формі та ширині розподілів підтверджують, що внутрішньокласова стабільність ембедингів може істотно залежати від мовця, навіть за однакових умов запису. На практиці це означає, що для деяких користувачів короткі командні висловлювання можуть бути менш надійними для автентифікації.

3.2. Порівняння коротких голосових команд із довгим еталонним голосовим відбитком

На наступному етапі дослідження було проаналізовано, наскільки ефективно короткі голосові команди можуть бути співвіднесені з еталонним голосовим відбитком, сформованим на основі довгих та акустично стабільних записів мовлення. Такий сценарій відтворює типову конфігурацію систем голосової автентифікації, де процес enrollment передбачає використання тривалого та якісного аудіосигналу, тоді як щоденні запити користувача представлені короткими командами тривалістю 1-1.5 секунди. У нашому випадку довгі еталонні записи було отримано з Dataset #2 (чоловічий голос, 15.78 с, Додаток А) та Dataset #4 (жіночий голос, 12.88 с, Додаток А), тоді як короткі команди взято відповідно з Dataset #1 і Dataset #3 (Додаток А).

Для кожного диктора було сформовано еталонний d-vector, після чого всі короткі командні висловлювання порівнювалися з цим відбитком за допомогою стандартної метрики косинусної подібності. Таким чином отримано по 17 значень для кожного диктора, що відображають ступінь відповідності коротких команд довгому «золотому стандарту» голосової ідентичності. Такий підхід дозволяє оцінити, наскільки надійно Resemblyzer може зіставляти короткий запит зі стійким, багатоканальним enrollment-профілем, який містить значно більше акустичної інформації. Результати порівняння представлені у вигляді діапазонів подібності і часток значень у кожному діапазоні (Табл. 3, 4, Рис. 3.3).

Як видно з цих таблиць та рисунку, в обох випадках жоден із коротких аудіосигналів не досяг порога успішної автентифікації при порівнянні з високоякісним еталонним відбитком. Переважна більшість значень перебувала у проміжному інтервалі (0.60-0.80), а значна частина – у зоні однозначної невдалої автентифікації (<0.60).

Таблиця 3. Порівняння даних Dataset #1 з Dataset #2 (чоловічий голос, короткі команди ↔ довгий еталонний відбиток)

Table 3. Comparison of data from Dataset #1 and Dataset #2 (male voice, short commands ↔ long reference embedding)

Діапазон	Частка (%)	Інтерпретація
[0;0.6]	17.65	Невдала автентифікація
(0.6;0.8)	82.35	Проміжний діапазон невизначеності
[0.8;1]	0.0	Успішна автентифікація
Всього зразків: 17		

Таблиця 4. Порівняння даних Dataset #3 з Dataset #4 (жіночий голос, короткі команди ↔ довгий еталонний відбиток)

Table 4. Comparison of data from Dataset #3 and Dataset #4 (female voice, short commands ↔ long reference embedding)

Діапазон	Частка (%)	Інтерпретація
[0;0.6]	35.29	Невдала автентифікація
(0.6;0.8)	64.71	Проміжний діапазон невизначеності
[0.8;1]	0.0	Успішна автентифікація
Всього зразків: 17		

Ці результати свідчать, що короткі команди тривалістю 1-1.5 секунди не містять достатнього обсягу стабільних індивідуальних акустичних ознак, необхідних для коректної верифікації при порівнянні з довгим записом.

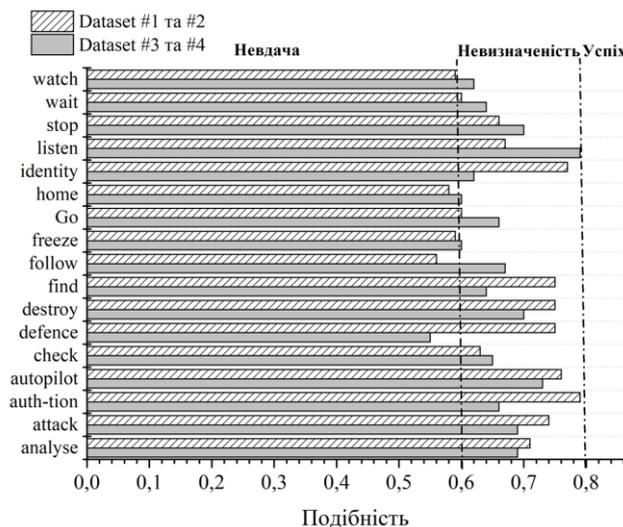


Рис. 3.3. Порівняння подібності коротких команд із довгим еталонним записом (чоловічий та жіночий голоси). Невдача, невизначеність та успіх означають діапазони подібностей, які відповідають невдалій автентифікації, проміжному діапазону невизначеності та успішній автентифікації, відповідно

Fig. 3.3. Comparison of similarity between short commands and a long reference recording (male and female voices). Failure, uncertainty, and success denote similarity ranges corresponding to unsuccessful authentication, an intermediate uncertainty region, and successful authentication, respectively)

Отримані результати можуть бути пояснені архітектурою Resemblyzer, що використовує LSTM-енкодер з усередненням прихованих станів на часовому горизонті всього запису. Короткі аудіофрагменти містять недостатню кількість інформаційно насичених фреймів, що призводить до нестабільності ембеддингу, підвищеної чутливості до випадкових інтонаційних варіацій та фонетичних коливань і, відповідно, – до низької подібності з довгим еталонним записом. У контексті IoT-застосувань це означає, що короткі командні висловлювання не можуть бути використані як самостійний матеріал для автентифікації, якщо enrollment здійснено на довгому аудіосигналі.

Описаний аналіз демонструє фундаментальну обмеженість Resemblyzer у сценаріях короткокомандної автентифікації: короткі команди тривалістю близько однієї секунди не є

достатньо інформативними для відтворення структури еталонного d-vector. Для практичного застосування такі результати вказують на необхідність використання або довших команд, або багатократної агрегації коротких фрагментів, або застосування спеціальних технік нормалізації та підсилення голосових ознак.

3.3. Аналіз якості автентифікації при збільшенні тривалості та розміру аудіозаписів

Останній етап дослідження було спрямовано на дослідження здатності Resemblyzer формувати стабільніші голосові відбитки та підвищувати точність автентифікації при збільшенні тривалості коротких команд та, відповідно, обсягу мовленнєвого матеріалу. Доцільність такого роду аналізу впливає з результатів попереднього підpunkту, де короткі фрагменти тривалістю приблизно 1 секунду демонстрували недостатню інформативність для надійного зіставлення з довгими enrollment-записами. Для перевірки цієї гіпотези було сформовано Dataset #5 (Додаток А), у якому кожна команда з Dataset #1 та Dataset #3 була перезаписана у вигляді довших та якісніших фрагментів. Як показано в Табл. В1, Додаток В, середній розмір файлу та середня тривалість збільшилися у ~2-2.4 рази. Це дозволило оцінити, чи є наявність додаткових мовленнєвих кадрів достатньою для формування більш стабільних голосових відбитків.

Спочатку було обчислено матрицю косинусної подібності для всіх аудіозразків Dataset #5 відносно один одного (Рис. 3.4). Результати виявили значне підвищення рівня внутрішньокласової стабільності порівняно з аналогічними матрицями для коротких команд: більшість значень лежала у діапазоні 0.85-0.95, що наближається до типових показників «високої подібності» для систем контролю доступу. Максимальні значення коливалися біля 0.94-0.95, а мінімальні – практично не виходили за межі 0.8.

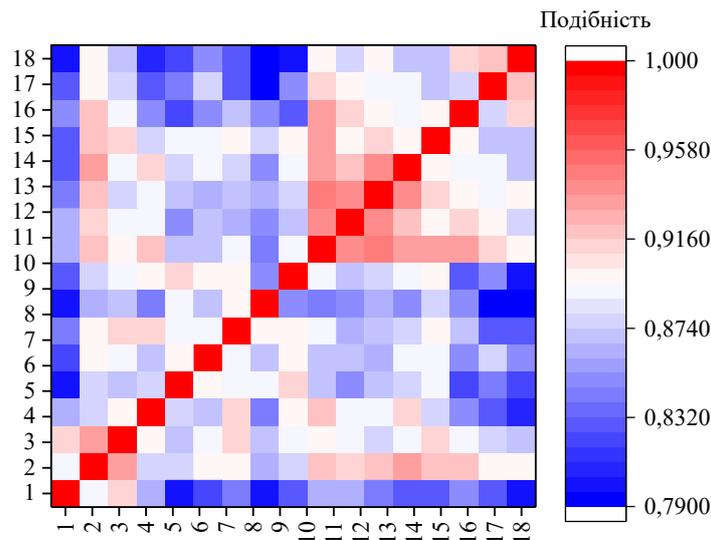


Рис. 3.4. Матриця косинусної подібності для всіх аудіозразків Dataset #5 відносно один одного. Номерами позначені команди, повна розшифровка яких представлена у Додатку В

Fig. 3.4. Cosine similarity matrix for all audio samples in Dataset #5 relative to each other. Command indices correspond to the full list provided in Appendix C.

Статистичний розподіл подібності всередині Dataset #5 показав, що 98.53% усіх пар лежать у діапазоні [0.8-1.0], тобто відповідають критерію «успішна автентифікація», 1.47% – у проміжному інтервалі (0.6-0.8) та 0% – у зоні невдалої автентифікації [0-0.6] (Рис. 3.5, штрихована заливка). Загальна кількість порівнянь становила 272, і серед них не було зафіксовано жодного хибно-позитивного результату. Це свідчить про те, що вже збільшені до 2.6–3.0 секунд аудіозаписи забезпечують утворення стабільних голосових відбитків, придатних для внутрішньокласового зіставлення.

Далі було оцінено здатність збільшених команд із Dataset #5 узгоджуватися з голосовим еталоном (d-vector), побудованим на основі високоякісного довгого аудіофрагмента із Dataset #2. Цей сценарій моделює реальну задачу автентифікації, коли enrollment виконується на основі

«еталонного» запису, а перевірка – за допомогою покращених, але все ще відносно коротких команд.

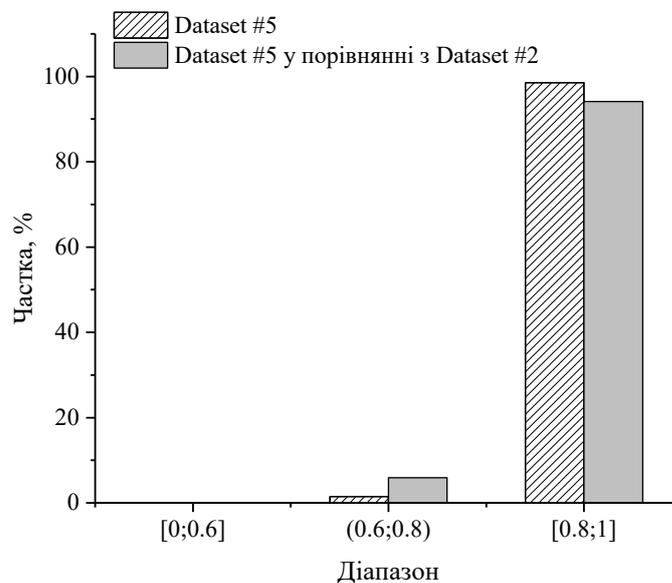


Рис. 3.5. Статистичний розподіл подібностей для Dataset #5 (штрихована заливка) та при порівнянні Dataset #5 з Dataset #2 (сіра заливка)

Fig. 3.5. Statistical distribution of similarity values for Dataset #5 (dashed shading) and for the comparison between Dataset #5 and Dataset #2 (gray shading)

Результати порівняння, представлені на Рис. 3.5 (сіра заливка) показали, що 94.12% команд успішно зіставилися з еталоном (подібність ≥ 0.8), 5.88% потрапили у зону невизначеності (0.6-0.8) та 0% не пройшли автентифікацію (≤ 0.6). Таким чином, покращені записи забезпечують рівень успішних збігів, який відповідає характеристикам повноцінних систем верифікації мовця, що використовують значно довші аудіосигнали.

Порівняння внутрішньокласових значень подібності для Dataset #5, а також їх зіставлення з еталоном із Dataset #2, демонструє узгоджену та чітко виражену тенденцію: збільшення тривалості аудіосигналу до приблизно 2.6-3.0 секунд істотно підвищує якість отриманих ембедингів. Довші записи забезпечують більшу кількість мовленнєвих кадрів, що, у свою чергу, призводить до формування ембедингів із вищою міжфреймовою узгодженістю та меншою варіативністю. Це дозволяє моделі Resemblyzer відтворювати індивідуальні голосові характеристики значно точніше, ніж у випадку коротких команд, зафіксованих у Dataset #1 та Dataset #3.

Внутрішньокласові значення косинусної подібності для Dataset #5 стабільно наближалися до діапазону високої відповідності, що традиційно вважається достатнім для надійної верифікації користувача. Така поведінка суттєво контрастує з результатами попередніх експериментів, де короткі записи не могли забезпечити порівнюваний рівень акустичної інформативності. Примітно, що збільшені командні записи зберігають високу подібність навіть у випадку співставлення з довгим еталоном голосовим профілем, що додатково підтверджує їхню придатність для практичних задач автентифікації.

Отримані результати дають підстави стверджувати, що короткі команди тривалістю близько однієї секунди є недостатніми для надійного розпізнавання користувача, тоді як фрагменти тривалістю близько трьох секунд вже формують d-vector ембединги достатньої якості для стабільного відтворення голосових характеристик. У прикладних сценаріях, пов'язаних з IoT та голосовим керуванням, це означає, що оптимальною стратегією є використання довгих enrollment-записів у поєднанні з командними аудіофрагментами середньої тривалості. Такий підхід забезпечує доцільний компроміс між зручністю користувача та вимогами до безпеки, зберігаючи при цьому можливість реалізації автентифікації без застосування високопродуктивних обчислювальних ресурсів.

4. Висновки

У даній роботі проведено комплексну оцінку можливостей бібліотеки Resemblyzer для задач голосової автентифікації на основі коротких аудіофрагментів у контексті автономних та ресурсно обмежених систем. Аналіз внутрішньокласової варіативності та порівняння коротких команд із довгими еталонними записами показали, що фрагменти тривалістю близько однієї секунди не забезпечують достатньої інформативності для надійної верифікації користувача. Це підтверджує обмеженість ультракоротких команд у біометричних системах та узгоджується з сучасними уявленнями про необхідність достатнього часово-спектрального покриття мовного сигналу для вилучення індивідуально-специфічних ознак. Натомість, збільшення тривалості команд до 2.6-3.0 секунд істотно покращує якість d-vector ембеддингів і забезпечує рівень подібності, який відповідає практичним вимогам систем контролю доступу та персональної ідентифікації. Отримані результати демонструють формування більш компактних і стабільних кластерів у векторному просторі ознак, що знижує внутрішньокласову дисперсію та підвищує надійність автентифікації.

Проведені дослідження також підтвердили, що Resemblyzer може ефективно працювати на пристроях із низькою обчислювальною потужністю без використання хмарних сервісів, що робить його перспективним інструментом для створення легких, автономних і безпечних голосових інтерфейсів. Подальші дослідження доцільно спрямувати на оцінку стійкості моделі в умовах акустичних завад, реверберації та фонового шуму, а також на вивчення впливу мовних факторів, таких як темп мовлення, емоційний стан та багатомовність. Окремий інтерес становить аналіз продуктивності підходу у багатокористувацьких IoT-сценаріях, де критичними є масштабованість, захист від спуфінгових атак і довготривала стабільність голосових шаблонів.

REFERENCES

1. A. Choudhary, Internet of Things: a comprehensive overview, architectures, applications, simulation tools, challenges and future directions. *Discov. Internet Things*. 2024. Vol. 4. P. 31. <https://doi.org/10.1007/s43926-024-00084-3>.
2. M. Lombardi, F. Pascale, D. Santaniello, Internet of Things: A General Overview between Architectures, Protocols and Applications. *Information*. 2021. Vol. 12. P. 87. <https://doi.org/10.3390/info12020087>.
3. L. Atzori, A. Iera, G. Morabito, The Internet of Things: A survey. *Computer Networks*. 2010. Vol. 54. P. 2787-2805. <https://doi.org/10.1016/j.comnet.2010.05.010>.
4. M. Hoy, Alexa, Siri, Cortana, and More: An Introduction to Voice Assistants. *Med. Ref. Serv. Q.* 2018. Vol. 37. P. 81-88. <https://doi.org/10.1080/02763869.2018.1404391>.
5. M. Benzeghiba, R. De Mori, O. Deroo et al., Automatic speech recognition and speech variability. *Speech Commun.* 2007. Vol. 49. P. 763-786. <https://doi.org/10.1016/j.specom.2007.02.006>.
6. A. Javed, K. Malik, H. Malik, A. Irtaza, Voice spoofing detector: a unified anti-spoofing framework. *Expert Systems Applic.* 2022. Vol. 198. P. 116770. <https://doi.org/10.1016/j.eswa.2022.116770>.
7. N. Ahmed, J. Khan, N. Sheta et al., Detecting Replay Attack on Voice-Controlled Systems using Small Neural Networks. *2022 IEEE 7th Forum on Research and Technologies for Society and Industry Innovation (RTSI)*, Paris, France. 2022. P. 50-54. <https://doi.org/10.1109/RTSI55261.2022.9905158>.
8. Z. Wu, N. Evans, T. Kinnunen et al., Spoofing and countermeasures for speaker verification: A survey. *Speech Commun.* 2015. Vol. 66. P. 130-153. <https://doi.org/10.1016/j.specom.2014.10.005>.
9. T. Kinnunen, Z. Wu, K. Lee et al., Vulnerability of speaker verification systems against voice conversion spoofing attacks: The case of telephone speech. *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Kyoto, Japan. 2012. P. 4401-4404. <https://doi.org/10.1109/ICASSP.2012.6288895>.
10. A. Poddar, M. Sahidullah, G. Saha, Speaker verification with short utterances: a review of challenges, trends and opportunities. *IET Biometrics*. 2018. Vol. 7. P. 403-411. <https://doi.org/10.1049/iet-bmt.2017.0065>.
11. N. Dehak, P. Kenny, R. Dehak et al., Front-End Factor Analysis for Speaker Verification. *IEEE Transactions on Audio, Speech, and Language Processing*. 2011. Vol. 19. P. 788-798. <https://doi.org/10.1109/TASL.2010.2064307>.
12. D. Snyder, D. Garcia-Romero, G. Sell et al., X-Vectors: Robust DNN Embeddings for Speaker Recognition. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, AB, Canada. 2018. P. 5329-5333. <https://doi.org/10.1109/ICASSP.2018.8461375>.

13. A. Baeovski, Y. Zhou, A. Mohamed, M. Auli, wav2vec 2.0: A framework for self-supervised learning of speech representations. *Adv. Neural Inform. Proces. Syst.* 2020. Vol. 33. P. 12449-12460. <https://doi.org/10.48550/arXiv.2006.11477>.
14. L. Wan, Q. Wang, A. Papir, I. Moreno, Generalized end-to-end loss for speaker verification. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2018. P. 4879-4883. <https://doi.org/10.48550/arXiv.1710.10467>.
15. M. Ravanelli, T. Parcollet, P. Plantinga et al., SpeechBrain: A general-purpose speech toolkit. *arXiv preprint arXiv:2106.04624*. 2021. <https://doi.org/10.48550/arXiv.2106.04624>.
16. H. Bredin, R. Yin, J. Coria, Pyannote. audio: neural building blocks for speaker diarization. *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2020. P. 7124-7128). <https://doi.org/10.48550/arXiv.1911.01255>.
17. Y. Jia, Y. Zhang, R. Weiss et al., Transfer learning from speaker verification to multispeaker text-to-speech synthesis. *Adv. Neur. Inform. Proces. Systems.* 2018. arXiv:1806.04558. <https://doi.org/10.48550/arXiv.1806.04558>.
18. Q. Wang, C. Downey, L. Wan et al., Speaker diarization with LSTM. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2018. P. 5239-5243. <https://doi.org/10.48550/arXiv.1710.10468>.

Додаток А

Повний опис експериментального масиву даних

Dataset #1. Тип голосу: чоловічий

Dataset #1. Voice type: male

File name	Text in audio	Length of audio (seconds)
Analyze.wav	Analyze	1,16
Attack.wav	Attack	1,34
Authentication.wav	Authentication	1,50
Autopilot.wav	Autopilot	1,42
Check.wav	Check	1,29
Defence.wav	Defence	1,48
Destroy.wav	Destroy	1,53
Find.wav	Find	1,46
Follow.wav	Follow	1,42
Freeze.wav	Freeze	1,40
Go.wav	Go	1,42
Home.wav	Home	1,34
Identity.wav	Identity	1,59
Listen.wav	Listen	1,20
Stop.wav	Stop	1,02
Wait.wav	Wait	1,18
Watch.wav	Watch	1,23

Dataset #2. Тип голосу: чоловічий

Dataset #2. Voice type: male

File name	Text in audio	Length of audio (seconds)
A1-m.wav	The Adventures of Sherlock Holmes" by Arthur Conan Doyle is a collection of detective stories written during the late 19th century. The book introduces the legendary detective Sherlock Holmes and his loyal companion, Dr. John Watson	15,78

Dataset #3. Тип голосу: жіночий

Dataset #3. Voice type: female

File name	Text in audio	Length of audio (seconds)
Analyze.wav	Analyze	1,49
Attack.wav	Attack	1,24
Authentication.wav	Authentication	1,84
Autopilot.wav	Autopilot	1,84
Check.wav	Check	1,19
Defence.wav	Defence	1,52
Destroy.wav	Destroy	1,33
Find.wav	Find	1,37
Follow.wav	Follow	1,38
Freeze.wav	Freeze	0,55
Go.wav	Go	1,2
Home.wav	Home	1,42
Identity.wav	Identity	1,48
Listen.wav	Listen	1,47
Stop.wav	Stop	1,15
Wait.wav	Wait	1,19
Watch.wav	Watch	1,46

Dataset #4. Тип голосу: жіночий

Dataset #4. Voice type: female

File name	Text in audio	Length of audio (seconds)
A1-f.wav	The Adventures of Sherlock Holmes" by Arthur Conan Doyle is a collection of detective stories written during the late 19th century. The book introduces the legendary detective Sherlock Holmes and his loyal companion, Dr. John Watson	12,88

Dataset #5. Тип голосу: чоловічий

Dataset #5. Voice type: male

File name	Text in audio	Length of audio (seconds)
Gltm-Analyze.wav	Ginger, listen to me, Analyze	3,16
Gltm-Attack.wav	Ginger, listen to me, Attack	3,07
Gltm-Authentication.wav	Ginger, listen to me, Authentication	3,64
Gltm-Autopilot.wav	Ginger, listen to me, Autopilot	3,02
Gltm-Check.wav	Ginger, listen to me, Check	3,21
Gltm-Defence.wav	Ginger, listen to me, Defence	3,03
Gltm-Destroy.wav	Ginger, listen to me, Destroy	3,16
Gltm-Find.wav	Ginger, listen to me, Find	2,68
Gltm-Follow.wav	Ginger, listen to me, Follow	3,89
Gltm-Freeze.wav	Ginger, listen to me, Freeze	2,86
Gltm-Go.wav	Ginger, listen to me, Go	2,63
Gltm-Home.wav	Ginger, listen to me, Home	2,73
Gltm-Identity.wav	Ginger, listen to me, Identity	2,92
Gltm-Listen.wav	Ginger, listen to me, Listen	2,95
Gltm-Stop.wav	Ginger, listen to me, Stop	2,78
Gltm-Wait.wav	Ginger, listen to me, Wait	2,68
Gltm-Watch.wav	Ginger, listen to me, Watch	2,86

Додаток Б

Таблиця Б1. Матриця косинусної подібності Dataset # 1

Table B1. Cosine similarity matrix for Dataset #1

	analyse	attack	auth-tion	autopilot	check	defence	destroy	find	follow	freeze	go	home	identity	listen	stop	wait	watch
analyse	1	0,74	0,72	0,65	0,67	0,79	0,69	0,8	0,81	0,78	0,79	0,85	0,79	0,86	0,72	0,83	0,8
attack	0,74	1	0,67	0,75	0,71	0,69	0,63	0,75	0,73	0,71	0,71	0,64	0,73	0,75	0,84	0,81	0,8
auth-tion	0,72	0,67	1	0,66	0,58	0,8	0,7	0,77	0,64	0,65	0,61	0,62	0,81	0,77	0,58	0,64	0,62
autopilot	0,65	0,75	0,66	1	0,62	0,63	0,67	0,71	0,75	0,67	0,69	0,64	0,71	0,6	0,7	0,7	0,7
check	0,67	0,71	0,58	0,62	1	0,57	0,65	0,72	0,78	0,76	0,81	0,67	0,62	0,64	0,74	0,81	0,75
defence	0,79	0,69	0,8	0,63	0,57	1	0,72	0,82	0,68	0,68	0,66	0,69	0,79	0,82	0,64	0,67	0,63
destroy	0,69	0,63	0,7	0,67	0,65	0,72	1	0,75	0,74	0,75	0,77	0,76	0,8	0,75	0,65	0,73	0,7
find	0,8	0,75	0,77	0,71	0,72	0,82	0,75	1	0,76	0,79	0,79	0,74	0,84	0,83	0,7	0,8	0,72
follow	0,81	0,73	0,64	0,75	0,78	0,68	0,74	0,76	1	0,82	0,89	0,82	0,71	0,73	0,77	0,87	0,83
freeze	0,78	0,71	0,65	0,67	0,76	0,68	0,75	0,79	0,82	1	0,86	0,84	0,81	0,81	0,78	0,86	0,84
go	0,79	0,71	0,61	0,69	0,81	0,66	0,77	0,79	0,89	0,86	1	0,87	0,75	0,72	0,82	0,88	0,83
home	0,85	0,64	0,62	0,64	0,67	0,69	0,76	0,74	0,82	0,84	0,87	1	0,73	0,8	0,76	0,85	0,83
identity	0,79	0,73	0,81	0,71	0,62	0,79	0,8	0,84	0,71	0,81	0,75	0,73	1	0,84	0,7	0,76	0,73
listen	0,86	0,75	0,77	0,6	0,64	0,82	0,75	0,83	0,73	0,81	0,72	0,8	0,84	1	0,68	0,79	0,79
stop	0,72	0,84	0,58	0,7	0,74	0,64	0,65	0,7	0,77	0,78	0,82	0,76	0,7	0,68	1	0,84	0,85
wait	0,83	0,81	0,64	0,7	0,81	0,67	0,73	0,8	0,87	0,86	0,88	0,85	0,76	0,79	0,84	1	0,93
watch	0,8	0,8	0,62	0,7	0,75	0,63	0,7	0,72	0,83	0,84	0,83	0,83	0,73	0,79	0,85	0,93	1

Таблиця Б2. Матриця косинусної подібності Dataset # 3

Table B2. Cosine similarity matrix for Dataset #3

	analyse	attack	auth-tion	autopilot	check	defence	destroy	find	follow	freeze	go	home	identity	listen	stop	wait	watch
analyse	1	0,8	0,78	0,8	0,76	0,79	0,79	0,87	0,72	0,72	0,67	0,71	0,77	0,78	0,76	0,73	0,75
attack	0,8	1	0,86	0,92	0,86	0,88	0,88	0,86	0,8	0,8	0,73	0,76	0,86	0,82	0,84	0,81	0,8
auth-tion	0,78	0,86	1	0,9	0,83	0,88	0,84	0,84	0,76	0,78	0,7	0,73	0,91	0,82	0,81	0,8	0,77
autopilot	0,8	0,92	0,9	1	0,8	0,87	0,84	0,85	0,79	0,8	0,71	0,73	0,86	0,81	0,82	0,81	0,79
check	0,76	0,86	0,83	0,8	1	0,86	0,87	0,8	0,87	0,86	0,8	0,86	0,84	0,86	0,88	0,89	0,87
defence	0,79	0,88	0,88	0,87	0,86	1	0,92	0,89	0,75	0,84	0,76	0,75	0,92	0,85	0,82	0,83	0,81
destroy	0,79	0,88	0,84	0,84	0,87	0,92	1	0,85	0,74	0,75	0,79	0,76	0,87	0,79	0,82	0,8	0,8
find	0,87	0,86	0,84	0,85	0,8	0,89	0,85	1	0,77	0,79	0,76	0,77	0,85	0,82	0,8	0,81	0,82
follow	0,72	0,8	0,76	0,79	0,87	0,75	0,74	0,77	1	0,83	0,85	0,9	0,72	0,83	0,84	0,85	0,84
freeze	0,72	0,8	0,78	0,8	0,86	0,84	0,75	0,79	0,83	1	0,74	0,8	0,82	0,87	0,81	0,89	0,87
go	0,67	0,73	0,7	0,71	0,8	0,76	0,79	0,76	0,85	0,74	1	0,88	0,72	0,76	0,8	0,79	0,83
home	0,71	0,76	0,73	0,73	0,86	0,75	0,76	0,77	0,9	0,8	0,88	1	0,73	0,8	0,88	0,85	0,86
identity	0,77	0,86	0,91	0,86	0,84	0,92	0,87	0,85	0,72	0,82	0,72	0,73	1	0,85	0,79	0,82	0,79
listen	0,78	0,82	0,82	0,81	0,86	0,85	0,79	0,82	0,83	0,87	0,76	0,8	0,85	1	0,84	0,9	0,83
stop	0,76	0,84	0,81	0,82	0,88	0,82	0,82	0,8	0,84	0,81	0,8	0,88	0,79	0,84	1	0,84	0,9
wait	0,73	0,81	0,8	0,81	0,89	0,83	0,8	0,81	0,85	0,89	0,79	0,85	0,82	0,9	0,84	1	0,87
watch	0,75	0,8	0,77	0,79	0,87	0,81	0,8	0,82	0,84	0,87	0,83	0,86	0,79	0,83	0,9	0,87	1

У таблицях Б1 та Б2 наведено матриці косинусної подібності між d-vector ембеддингами коротких голосових команд для Dataset #1 (чоловічий голос) та Dataset #3 (жіночий голос) відповідно. Наведені значення дозволяють кількісно оцінити внутрішньокласову варіативність мовних сигналів та стабільність формування голосових представлень. Порівняльний аналіз демонструє високий рівень внутрішньокласової подібності в обох випадках, переважно в діапазоні 0.75-0.90, що свідчить про здатність моделі формувати узгоджені голосові ембеддинги для коротких мовних фрагментів. Для жіночого голосу (Dataset #3) спостерігається дещо вища середня подібність і менша дисперсія значень, що може бути пов'язано з більш стабільними спектральними характеристиками. Разом із тим, наявність окремих комбінацій команд із пониженими значеннями подібності ($\approx 0.65-0.75$) відображає вплив фонетичної структури та артикуляційної варіабельності, що необхідно враховувати під час вибору порогів прийняття рішень у системах голосової автентифікації.

Додаток В

Таблиця В1. Фізичні характеристики dataset #5

Table C1. Acoustic characteristics of Dataset #5

	Dataset #1. Розмір файла (байти)	Dataset #3. Розмір файла (байти)	Dataset #5 Розмір файла (байти)	Середній фактор збільшення розміру файла	Dataset #1. Довжина аудіозапису (сек)	Dataset #3. Довжина аудіозапису (сек)	Dataset #5 Довжина аудіозапису (сек)	Середній фактор збільшення довжини аудіозапису
analyse	222802	286162	606802	2.42	1.16	1.49	3.16	2.42
attack	257362	238162	589522	2.38	1.34	1.24	3.07	2.38
auth-tion	288082	353362	698962	2.20	1.50	1.84	3.64	2.20
autopilot	272722	353362	579922	1.88	1.42	1.84	3.02	1.88
check	247762	228562	616402	2.59	1.29	1.19	3.21	2.59
defence	284242	291922	581842	2.02	1.48	1.52	3.03	2.02
destroy	293842	255442	606802	2.22	1.53	1.33	3.16	2.22
find	280402	263122	514642	1.89	1.46	1.37	2.68	1.89
follow	272722	265042	554962	2.06	1.42	1.38	3.89	2.06
freeze	268882	247762	549202	2.12	1.40	0.55	2.86	2.12
go	272722	230482	505042	2.02	1.42	1.2	2.63	2.02
home	257362	272722	524242	1.97	1.34	1.42	2.73	1.97
identity	305362	284242	560722	1.90	1.59	1.48	2.92	1.90
listen	230482	282322	566482	2.23	1.20	1.47	2.95	2.23
stop	195922	220882	533842	2.57	1.02	1.15	2.78	2.57
wait	226642	228562	514642	2.26	1.18	1.19	2.68	2.26
watch	236242	280402	549202	2.14	1.23	1.46	2.86	2.14

Trusov Mykhaylo *PhD student, V. N. Karazin Kharkiv National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine*

Turuta Oleksiy *Associate Professor of the Department of Program Engineering, Kharkiv National University of Radioelectronics, 14 Nauky Ave., Kharkiv 61166, Ukraine*

Uzlov Dmitro *Associate Professor of the Department of Theoretical and Applied Informatics, V. N. Karazin Kharkiv National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine*

Analysis of the effectiveness of the Resemblyzer library for short-command voice authentication

Relevance. Voice interaction is widely used in Internet of Things systems and autonomous embedded devices. However, its practical deployment is constrained by security and privacy requirements as well as the limited computational resources of edge platforms. This creates a demand for fully local voice authentication solutions capable of operating without reliance on cloud services. **Goal.** The objective of this study is to evaluate the capabilities of the open-source Python library Resemblyzer for implementing autonomous user voice authentication based on short voice commands under conditions of no access to cloud computing and limited hardware resources. **Research methods.** The study was conducted using several audio datasets with varying duration, quality, and file size. Voice embeddings generated by the Resemblyzer library were used for feature representation. Quantitative similarity assessment between recordings was performed using the cosine similarity metric in scenarios involving comparisons of recordings from the same speaker and from different speakers.

Results. The results demonstrate that reliable voice authentication is achieved for audio recordings with a duration of at least 2.63 seconds and a file size of no less than 495 kB. Short fragments with durations of 1-1.5 seconds were found to be insufficiently informative for stable speaker discrimination, particularly when compared against a high-quality reference recording. A clear dependence of authentication performance on the amount of acoustic information contained in the voice signal was identified.

Conclusions. The obtained results confirm the applicability of Resemblyzer for the development of fully autonomous real-time voice biometric authentication systems. Practical requirements for the minimum duration and informational richness of voice commands are formulated, which may be interpreted as technical constraints on the entropy of voice passwords in secure IoT applications.

Keywords: *voice control, user authentication, Resemblyzer, short voice commands, Internet of Things (IoT), resource-constrained devices, voice verification, voice embedding, cosine similarity.*

УДК (UDC) 004.89

**Тюрдьо
Іван Миколайович**

аспірант факультету математики і інформатики, кафедра
прикладної математики
Харківський національний університет імені В. Н. Каразіна, майдан
Свободи, 4, м. Харків, 61022
e-mail: ivan.turdio@karazin.ua
<https://orcid.org/0009-0001-7315-3628>

**Кізілова
Наталія Миколаївна**

доктор фізико-математичних наук, професор кафедри прикладної
математики
Харківський національний університет імені В. Н. Каразіна, майдан
Свободи, 4, м. Харків, 61022
e-mail: kizilova@karazin.ua
<https://orcid.org/0000-0001-9981-7616>

Використання методів машинного навчання в сучасній математичній онкології

Мета роботи: виконати аналіз сучасних підходів до оцінки ефективності та безпеки препаратів антиракової терапії із застосуванням методів машинного навчання, а також визначити перспективи їхнього використання у сучасній математичній онкології, в онкологічних дослідженнях, які широко використовують математичне моделювання та комп'ютерні симуляції.

Методи дослідження: пошук та аналіз сучасних наукових публікацій, які стосуються тематики використання машинного навчання в онкології.

В **результаті** дослідження було зроблено систематичний огляд літератури за тематикою використання машинного навчання в онкології. Було виконано аналіз використання основних методів машинного навчання, таких як метод контрольованого навчання (Supervised Learning, SL), метод неконтрольованого навчання (Unsupervised Learning, UL) та метод навчання з підкріпленням (Reinforcement Learning, RL) у сучасній онкології. Наведено приклади використання різноманітних алгоритмів машинного навчання у дослідженнях, які пов'язані з антираковою терапією та онкологією взагалі. Проаналізовані переваги та недоліки використаних алгоритмів машинного навчання в залежності від задач, які потрібно вирішити.

Висновки: Методи машинного навчання вже досить широко використовуються в медичних дослідженнях в сфері онкології. Вони успішно застосовувались для вирішення багатьох питань та показували гарні результати, але існує ще багато напрямків в онкології в яких використання методів машинного навчання може принести значний вклад в покращення медичних досліджень та медичної допомоги при лікуванні онкологічних захворювань. Наприклад, дуже перспективним виглядає використання алгоритмів, які засновані на навчанні з підкріпленням в персоналізованій прецензійній медицині, методи якої займають значну роль в персоналізованому лікуванні онкологічних захворювань.

Ключові слова: машинне навчання, контрольоване навчання, неконтрольоване навчання, навчання з підкріпленням, онкологія, математична онкологія, прецизійна медицина, персоналізована терапія, медичні дослідження.

Як цитувати: Тюрдьо І. М., Кізілова Н. М. Використання методів машинного навчання в сучасній математичній онкології. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління*. 2025. вип. 68. С.98-112. <https://doi.org/10.26565/2304-6201-2025-68-10>

How to quote: I. Turdjo and N. Kizilova "The use of machine learning methods in modern mathematical oncology", *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 68, pp. 98-112, 2025. <https://doi.org/10.26565/2304-6201-2025-68-10> [in Ukrainian]

Вступ

Онкологічні захворювання залишаються однією з головних причин смертності у світі, попри значний прогрес у діагностиці та лікуванні. Рак класифікується як генетичне захворювання, яке розвивається внаслідок змін у генах, що контролюють функції клітин, особливо їхній ріст і поділ. У 2022 році в усьому світі було зафіксовано близько 20 мільйонів нових випадків онкологічних захворювань, а кількість смертей від раку сягнула 9,7 мільйона. За оцінками, 53,5 мільйона людей

залишалися живими через п'ять років після встановлення діагнозу [1]. Протягом життя приблизно кожна п'ята людина стикається з онкологічним захворюванням, при цьому кожен дев'ятий чоловік і кожна дванадцята жінка помирають від раку [2]. Через недостатність точних та неінвазивних маркерів виявлення раку часто є упередженим і не завжди правильним [3]. Незважаючи на величезні зусилля, витрачені на розробку хіміотерапії раку, часто ці методи лікування ефективні лише у відносно невеликій частині онкологічних пацієнтів. Хоча давно визнано, що ефективність протипухлинних препаратів може значно відрізнятись у окремих пацієнтів, одні й ті ж протоколи часто застосовуються без урахування різних характеристик ракових клітин кожного пацієнта, незважаючи на можливість того, що лікування може бути невідповідним [4].

Створення безпечних і ефективних ліків — це складний, затратний і тривалий процес. Згідно з даними Tufts Center of the Study of Drug Development, повний цикл розробки нового препарату триває понад десять років і коштує більше ніж 2,6 мільярда доларів [5]. Цей процес включає етапи відкриття, доклінічних досліджень і чотири фази клінічних випробувань, що завершуються затвердженням препарату регуляторними органами.

Однією з ключових проблем у сфері охорони здоров'я є своєчасне виявлення та мінімізація побічних реакцій на лікарські засоби (Adverse Drug Reactions, ADR) і медикаментозної токсичності. Попри жорстке регулювання та багаторівневі клінічні дослідження, ADR і токсичні ефекти препаратів залишаються провідними чинниками захворюваності й смертності, призводячи до госпіталізацій і високих витрат на медичне обслуговування [6]. Традиційні підходи до виявлення ризиків, як-от постмаркетинговий нагляд і фіксація побічних явищ, зазвичай є реактивними і не завжди дозволяють своєчасно виявити загрозові для здоров'я ефекти до того, як вони позначаються на великій кількості пацієнтів [7].

Пацієнти з онкологічними захворюваннями часто проходять лікування із застосуванням декількох лікарських засобів одночасно, що значно підвищує ризик розвитку побічних реакцій (ADR). Основною причиною цього є підвищена ймовірність лікарських взаємодій. Зокрема, ретроспективне дослідження показало, що зі 100 пацієнтів з онкологією, які були госпіталізовані до стаціонару, близько двох третин отримували щонайменше одну потенційно небезпечну комбінацію препаратів [8].

Наслідки ADR мають широкий спектр проявів. У клінічному аспекті вони можуть призвести до суттєвого погіршення стану здоров'я пацієнта, а в деяких випадках – навіть до смерті. Економічно це тягне за собою суттєве зростання витрат на медичне обслуговування, які пов'язані із додатковими госпіталізаціями, подальшим лікуванням і втратою працездатності [9].

Крім того, незаплановане звернення за медичною допомогою через ADR може негативно впливати як на фізичний, так і на психоемоційний стан пацієнтів. Це, у свою чергу, створює додаткове навантаження на систему охорони здоров'я, що може знижувати якість надання медичних послуг.

Тому пошук ефективних підходів до раннього виявлення та зниження ризиків ADR і медикаментозної токсичності є пріоритетом як у фармацевтичних дослідженнях, так і в системі охорони здоров'я. Покращення процесів виявлення небажаних реакцій на ранніх етапах може суттєво зменшити шкоду для пацієнтів, підвищити рівень їх безпеки, оптимізувати витрати на лікування і сприяти більш ефективній розробці нових лікарських засобів [7]. Для вирішення таких задач використовують бази даних доклінічних зразків, таких як клітинні лінії, органіди та ксенотрансплантати мишей, які дають детальні молекулярні профілі («оміки») мутованих генів даного пацієнта, так і реакції клітин пухлини на різні препарати. Крім того, використовуються зображення пухлини у динаміці за допомогою МРТ або КТ візуалізації [10].

Для обробки таких великих різноманітних масивів даних методи машинного навчання (Machine Learning, ML) відкривають нові перспективи для вирішення цих проблем в сфері математичної онкології, тобто в онкологічних дослідженнях які широко використовують математичне моделювання та симуляції [11]. Використання штучного інтелекту (ШІ) та аналізу великих даних дозволяє виявляти приховані закономірності, прогнозувати результати лікування та визначати фактори, що впливають на ефективність терапії. Ці методи демонструють високий потенціал у покращенні процесу розробки лікарських засобів і оптимізації їх застосування в клінічній практиці. Хоча огляд математичних методів і чисельних алгоритмів був темою низки публікацій [12, 13], стрімка динаміка розроблення і оновлення як баз даних вимірювань, так і

нових знань розвинення онкологічних захворювань і математичних методів розпізнавання і класифікації, викликає необхідність більш детального систематичного аналізу.

Метою даної статті є аналіз сучасних підходів до оцінки ефективності та безпеки препаратів антиракової терапії із застосуванням методів машинного навчання, а також визначення перспектив їхнього використання.

1. Підходи машинного навчання в сучасній онкології

Існують два ключові підходи до прогнозування ADR, які базуються на даних про пацієнта та його історію лікування: це методи статистичного аналізу виживання та моделі контрольованого машинного навчання. Для прогнозування виживання пацієнта часто застосовуються такі методи, як оцінка кривих виживаності за Капланом–Мейєром, а також модель пропорційних ризиків Кокса, яка дозволяє проводити часовий аналіз ймовірності виникнення побічних ефектів [14]. Попри широке використання, модель Кокса має низку обмежень, оскільки базується на низці жорстких припущень: лінійність, незалежність змінних, пропорційність ризиків тощо, – які не завжди дотримуються у реальних клінічних даних. Крім того, ця модель не може ефективно працювати з різномірними типами даних, такими як медичні зображення та генетична інформація.

Методи глибокого навчання (Deep Learning, DL) використовувались для комп'ютерної розробки ліків [15], для прогнозування протипухлинних пептидів на основі багатовидової екстракції ознак [16, 17], для раннього виявлення побічних реакцій на ліки та ступеню їх токсичності [7, 18], для передбачення антипухлинної дії обраного препарату [6, 19] або комбінацій препаратів [13, 20, 21, 22, 23] і їх синергетичної взаємодії [24] та можливого часу початку побічної реакції [25]. Особливо ефективними є моделі контрольованого машинного навчання, зокрема методи випадкових лісів (Random Forest), опорних векторних машин (Support Vector Machines, SVM) і штучних нейронних мереж (Artificial Neural Networks, ANN). Вони здатні аналізувати складні залежності та адаптуватися до гетерогенних даних завдяки використанню різних алгоритмічних підходів [26].

Моделі SVM ефективно навчаються та можуть використовувати нелінійні ядра (нелінійні функції перетворення). SVM успішно використовували для класифікації зразків тканин пухлин на чутливі до хіміотерапії або стійкі. SVM-регресія також використовувалася для пріоритизації ліків та прогнозування реакції пухлини [27]. Вибір функції ядра та інтерпретація SVM зазвичай не є простими задачами.

Одними з простих та інтуїтивно зрозумілих методів класифікації та регресії є методи найближчих сусідів. У методі k найближчих сусідів (k -Nearest Neighbors, KNN) вихідні дані для кожного нового запису з бази даних вимірювань базуються на вихідних даних меншої кількості k молекулярних профілів, які найбільш схожі на профіль нового зразка [28]. Метод KNN використовувався для розв'язання задач передбачення реакції на ліки (Drug Response Prediction, DRP), де чутливість нового зразка до ліків розраховувалася як середнє значення спостережуваних значень чутливості його найближчих сусідів [29]. Був досліджений вплив 453 препаратів хіміотерапії, а результати моделювання були застосовані до ~17 000 зразків РНК-секвенування з Атласу геному раку (The Cancer Genome Atlas) та бази даних експресії генотипу та тканин (Genotype-Tissue Expression). За допомогою методу KNN були виявлені багато відомих взаємодій між препаратами та генами, а також кілька потенційно нових асоціацій між ними. Однак методи найближчих сусідів часто проявляють ненавмисну та, як правило, невідому присутність багатьох нерелевантних ознак даних, а також вони є обчислювально ресурсоемними [12].

Загалом усі методи машинного навчання можна умовно поділити на три великі категорії: контрольоване навчання (Supervised Learning, SL), неконтрольоване навчання (Unsupervised Learning, UL) та навчання з підкріпленням (Reinforcement Learning, RL) (рис. 1) [30, 44]. Контрольоване навчання передбачає використання навчального набору з відомими вихідними значеннями для побудови предиктивної моделі, яка може бути застосована до нових, невідомих даних. Неконтрольоване навчання, у свою чергу, працює з немаркованими даними для виявлення прихованих структур, кластерів або закономірностей, і широко використовується, наприклад, у біоінформатиці. Алгоритми навчання з підкріпленням базуються на принципі послідовного прийняття рішень: модель шляхом проб і помилок навчається, які дії призводять до найкращих результатів у певному середовищі [31]. Нижче детальніше розглянуто особливості кожного з підходів.



Рис. 1 Основні класи машинного навчання

Fig. 1 Basic classes of machine learning

1.1 Контрольовані методи.

Алгоритми контрольованого навчання ґрунтуються на попередньо наданих знаннях і, по суті, відповідають на запитання типу: «враховуючи, що вже відомо...». Завдяки тому, що вони базуються на заздалегідь підготовленому навчальному наборі даних, ці методи є надзвичайно потужними для передбачення нових біологічних механізмів, специфічних властивостей клітинних ліній чи потенційних взаємодій. Використовуючи відому біологічну інформацію, контрольовані моделі здатні краще справлятися з фоновим шумом, характерним для великих біомедичних даних, і тому добре підходять для інтегративного аналізу, що охоплює численні джерела інформації.

Передумови для навчання таких алгоритмів зазвичай формуються на основі так званого «золотого стандарту», який являє собою еталонний набір прикладів, який ідеально (або майже ідеально) відображає досліджуване явище [33]. Проте, у контексті ракових захворювань наявні знання часто є неповними або неточними, що вимагає від алгоритмів здатності бути стійкими до таких недоліків у вхідних даних.

Контрольовані моделі можуть виконувати класифікацію, тобто побудову гіпотетичних правил, які дозволяють відрізнити об'єкти (наприклад, гени) різних класів. Найчастіше для таких завдань застосовуються метод SVM [35] та штрафна логістична регресія [36]. Золотий стандарт у цьому випадку складається з позитивних прикладів (які алгоритм має навчитися виявляти) та негативних прикладів (яких слід уникати).

Як логістична регресія, так і SVM прагнуть знайти просту межу між позитивними та негативними прикладами. Ступінь спрощення моделі регулюється параметром, що задається користувачем і балансує між точністю класифікації та складністю моделі. Простота моделі часто вважається перевагою, оскільки забезпечує кращу узагальнюваність результатів на нових даних.

Ключовим елементом SVM є функції ядра, які використовуються для відображення вхідних даних у простір вищої розмірності, що дозволяє виділити нелінійні залежності. Ці функції можуть бути різними за типом — лінійними, нелінійними, радіальними, поліноміальними тощо. Метод опорних векторів широко застосовується не лише для задач класифікації та регресії, але також адаптується до неконтрольованого навчання.

Крім класифікації, контрольоване навчання може виконувати регресійні задачі, тобто моделювання неперервних залежностей. У таких випадках найчастіше використовуються методи лінійної регресії або регресії на основі опорних векторів. Лінія найкращого підходу за методом

найменших квадратів, застосована до багатьох вимірювань, є прикладом моделі, побудованої за допомогою контрольованої регресії [33].

Алгоритми лінійної регресії прості і є найбільш популярними алгоритмами машинного навчання з широким спектром застосування. Лінійна модель, яка лежить в основі лінійної регресії, була основою статистики протягом останніх 30 років і залишається одним з наших найважливіших інструментів [28]:

$$f(x) = \beta_0 + \sum_{j=1}^p x_j \beta_j \quad (1)$$

де,

$f(x)$ – передбачене (прогнозоване) значення залежної змінної.

β_0 – вільний член (зсув).

x_j – значення j -ої незалежної змінної.

β_j – коефіцієнт при j -ій змінній.

p – кількість незалежних змінних.

Стандартний алгоритм, регресія найменших квадратів, використовує суму квадратичних залишків як функцію витрат, яку потрібно мінімізувати. Регресія найменших квадратів працює з простим набором даних; однак зі зростанням складності алгоритм показує перенавчання (низьке зміщення, але велика дисперсія). Для вирішення цієї проблеми використовується кілька алгоритмів, таких як модель гребня (ridge model), модель ласо (lasso model) та еластична сітка (elastic net) [37].

Серед різноманітних методів контрольованого навчання одним із найбільш застосовуваних для задач класифікації є дерево рішень (Decision Tree) – інтуїтивно зрозумілий підхід до побудови прогнозовної моделі, який ефективно працює з простими структурами даних. Модель починається з кореневого вузла, де обробляється весь набір даних, а далі послідовно розгалужується за допомогою логічних умов (гілок), доки не буде досягнуто остаточного результату у листовому вузлу. Кожен внутрішній вузол дерева відповідає певному атрибуту, що визначає подальший розподіл даних [38].

Такий підхід називається моделлю білої скриньки, оскільки процес прийняття рішень легко простежити, візуалізувати та інтерпретувати. Проте зі зростанням обсягу та складності вхідних даних дерева рішень схильні до перенавчання, що знижує їхню узагальнюючу здатність і точність прогнозування. Для подолання цього недоліку були розроблені удосконалені варіанти моделей, які базуються на поєднанні кількох дерев рішень.

Такі методи, що об'єднують декілька слабких моделей у сильну комбіновану модель, називаються ансамблевими (Ensemble models). Вони покликані забезпечити високу точність, стабільність та здатність до узагальнення. До таких моделей належать, зокрема, Random Forest та Gradient Boosting Machines (GBM).

У моделі Random Forest навчальна вибірка багаторазово розбивається випадковим чином (bootstrapping) на кілька підмножин, для кожної з яких будується окреме дерево. Використання декількох випадкових дерев рішень для прогнозування підвищує точність прогнозування [40].

Окрім паралельних ансамблів, існують також послідовні ансамблеві методи, зокрема адаптивне підсилення (AdaBoost) та градієнтне підсилення (Gradient Boosting). У AdaBoost для кожної ітерації використовується слабка модель (зазвичай дерево з одним вузлом), а кожна наступна модель фокусується на прикладах, що були класифіковані з помилкою. Градієнтне підсилення використовує дерева більшої глибини (зазвичай 8–32 листові вузли) для оптимізації похідної функції втрат. Обидва методи зазвичай демонструють вищу ефективність порівняно з окремими деревами [37].

Попри високу ефективність, ансамблеві підходи втрачають прозорість: через велику кількість об'єднаних дерев стає складно відстежити вплив конкретних вхідних параметрів на результат. Це ускладнює їх використання в критичних прикладних сферах, де необхідна повна інтерпретованість моделей, на відміну від лінійних моделей або окремих дерев рішень, що забезпечують чітке пояснення на основі вагових коефіцієнтів або гілок [41].

Іншими прикладами алгоритмів які використовуються в онкології та належать до контрольованого навчання є ANN. Ці алгоритми більш складні і використовуються на великих наборах даних з великою кількістю ознак, наприклад, для прогнозування відповіді на хіміотерапію по тисячі змінних. В найпростішому випадку [28,46] формула для одного шару ANN має вигляд

$$y = \sigma \left(\sum_{j=1}^n \omega_j x_j + b \right) \quad (2)$$

де,

x_j – вхідні значення (ознаки),

ω_j – вагові коефіцієнти,

b – зсув (bias),

σ – активаційна функція,

y – вихід нейрона (передбачення).

ANN встановлюють зв'язок між вхідними ознаками та прогнозованими результатами через багаторівневу структуру математичних перетворень. Модель відображає вхідні функції на вузли в прихованому шарі за допомогою лінійних функцій. Потім ці вузли відображають результат за допомогою нелінійної функції активації. Ця мережева динаміка дозволяє нейронним мережам фіксувати складні взаємодії між функціями та результатом. В останні роки було досягнуто значного прогресу в нейронних мережах, включаючи впровадження рекурентних нейронних мереж (Recurrent Neural Networks, RNN), згорткових нейронних мереж (Convolutional Neural Networks, CNN) і генеративних змагальних мереж (Generative Adversarial Networks, GAN). Нейронні мережі стали особливо популярними завдяки своїй здатності синтезувати необроблені зображення та вільний текст. Вони піддаються неструктурованим форматам даних і можуть масштабуватися до параметрів високої розмірності, а саме до випадків, коли кількість вхідних функцій значно перевищує кількість спостережень. Однак сила і складність моделювання досягаються за рахунок можливості інтерпретації. Нейронні мережі були названі методами чорної скриньки в результаті труднощів з отриманням інформації. Як і у випадку з ансамблевими методами, відсутність можливості інтерпретації обмежує їх корисність у певних клінічних умовах [41].

1.2 Неконтрольоване навчання

Неконтрольоване навчання передбачає роботу з вхідними даними, які не мають попередньо визначених міток або класів. Основною метою цього підходу є виявлення прихованих закономірностей, структур або угруповань у даних без чіткого нагляду чи цільової змінної. Після ідентифікації таких структур, наприклад кластерів, отриману інформацію можна використати для подальшої класифікації за допомогою контрольованих методів. У такому разі новий зразок може бути віднесений до найбільш подібного кластера [34].

Цей підхід дозволяє дослідити організацію даних, ставлячи запитання на кшталт: «Які закономірності виявляються у профілях експресії генів у пацієнтів з онкологічними захворюваннями?» Неконтрольовані алгоритми часто виявляють домінуючі і повторювані характеристики в даних, хоча й можуть бути чутливими до впливу зовнішніх або змішувальних факторів, які не є частиною основної структури даних [34].

На відміну від контрольованих методів, які орієнтовані на конкретні передбачувані результати, навчання без учителя має більш загальну дослідницьку спрямованість. Воно не базується на наявності заданої цільової змінної, як виживаність або відповідь на лікування, і натомість дає змогу ідентифікувати внутрішні закономірності в наборах даних.

Серед різних технік неконтрольованого навчання кластеризація є однією з найбільш інтерпретованих у контексті медицини, зокрема у клінічній інформатиці та біомедичних дослідженнях. Наприклад, кластерний аналіз електронних медичних записів (Electronic Medical Recordings, EMR) для пацієнтів з певним захворюванням може допомогти виявити різні підтипи пацієнтів або клінічні фенотипи в межах однієї діагностичної категорії.

Загалом алгоритми кластеризації розділяють дані на K кластерів, орієнтуючись на максимізацію внутрішньої однорідності (схожості між елементами в одному кластері) і

зовнішньої відмінності (між різними кластерами). Подібність між об'єктами зазвичай оцінюється на основі метрик відстані або схожості, таких як евклідова відстань чи кореляція [41].

Метою алгоритму К-середніх є мінімізувати функцію, яка обчислює суму квадратів відстаней всіх точок до центрів відповідних кластерів, тобто знайти такі кластери, де точки якомога ближчі до своїх центрів [44]:

$$J = \sum_{k=1}^k \sum_{i \in C_k} \|x_i - \mu_k\|^2 \quad (3)$$

де,

J – загальна сумарна квадратична помилка,

K – кількість кластерів,

C_k – множина точок, які належать до кластера k ,

x_i – об'єкт даних,

μ_k – центр (середнє) кластера k ,

$\|x_i - \mu_k\|^2$ – евклідова відстань (в квадраті) між точкою та центром її кластера.

Задача ідентифікації молекулярних підтипів зляжисних новоутворень є типовим прикладом проблеми, яку доцільно вирішувати за допомогою неконтрольованих методів аналізу. За наявності даних експресії генів для різних зразків пухлин дослідник прагне з'ясувати, чи існують спільні моделі експресії, які дозволяють згрупувати ці зразки у підтипи на основі їх молекулярних характеристик. Для цього часто використовуються алгоритми кластеризації, зокрема ті, що передбачають поділ даних на заздалегідь задану кількість груп.

Одним із найбільш поширених методів у цій категорії є кластеризація k -середніх (k -means), яка виконує розподіл даних на кластери, групуючи спостереження з подібними характеристиками експресії. K -means кластеризація намагається розділити вимірювання на групи, які демонструють подібні закономірності [34]. Наприклад, у дослідженні [39] було продемонстровано ефективне застосування методу k -середніх для ідентифікації молекулярних підтипів раку яєчників, що корелювали з клінічними результатами. Якість неконтрольованої кластеризації зазвичай оцінюється за відтворюваністю отриманих підтипів у незалежних вибірках.

Серед найуживаніших кластерних методів — кластеризація k -середніх та ієрархічна кластеризація. Перший метод використовує евристичні підходи для знаходження оптимального розподілу даних на k кластерів, тоді як другий починає з того, що кожне спостереження вважається окремим кластером, і поступово об'єднує їх у більші групи на основі зростання міжкластерної відстані. Результатом є деревоподібна структура, що дозволяє досліднику вибрати бажану кількість кластерів на будь-якому рівні агрегації [41].

Інтерпретація кластерів є центральною проблемою в навчанні без учителя, особливо з огляду на актуальність кластеризації в дослідницькому аналізі даних [41].

1.3 Навчання з підкріпленням

Клінічне прийняття рішень не завжди є лінійним або залежним лише від одного чинника — лікарям часто доводиться ухвалювати терапевтичні рішення на основі неповної, фрагментарної чи неоднозначної інформації. Для подолання цих викликів у сфері охорони здоров'я зростає інтерес до методів навчання з підкріпленням RL — підрозділу машинного навчання, у якому інтелектуальний агент взаємодіє із середовищем, виконуючи дії згідно з певною стратегією з метою оптимізації функції винагороди.

У рамках RL агент розробляє політику — адаптивну стратегію дій, спрямовану на максимізацію сукупної винагороди. Цей процес, як правило, здійснюється методом ітеративного вдосконалення через проби і помилки, доки не буде наближено або досягнуто оптимальне рішення. Алгоритмічна реалізація RL часто базується на моделюванні марковського процесу прийняття рішень (Markov Decision Process, MDP), у якому визначаються змінні для станів, дій, функції винагороди та коефіцієнта дисконтування, що зменшує значення віддалених у часі наслідків [30, 45].

Кожна дія, вчинена агентом, змінює стан середовища — у контексті медицини це може стосуватись клінічного стану пацієнта чи супутніх даних. Залежно від ефекту, який має дія (наприклад, позитивна відповідь на терапію чи ускладнення), агент отримує винагороду або

покарання. Таким чином, агент формує політику, базуючись або на відгуках винагород, або на законах переходів між парами "стан–дія", які регулюють подальший вибір дій. Наприклад, агент може виявити, що підвищення дози хіміотерапії посилює протипухлинну відповідь, але також супроводжується збільшенням токсичності.

Загальна мета агента полягає не в досягненні максимальної миттєвої винагороди, а у максимізації довгострокової вигоди через обґрунтований вибір дій для кожного клінічного стану. Такий підхід дозволяє моделювати динамічні процеси прийняття рішень, що є особливо цінним у складних терапевтичних сценаріях [43].

Для оцінки ефективності політики використовують функцію значення стану $V^\pi(s)$ [45]:

$$V^\pi(s) = \sum_{a_t \in A} \pi(a_t | s_t) \sum_{s_{t+1} \in S} T(s_{t+1} | s_t, a_t) [R(s_t, s_{t+1}) + \gamma V^\pi(s_{t+1})] \quad (4)$$

де,

$V^\pi(s)$ – функція цінності стану s при політиці π . Вона показує очікувану сумарну нагороду, починаючи зі стану s і дотримуючись політики π .

$\pi(a_t | s_t)$ – ймовірність вибору дії a_t у стані s_t за політикою π .

$T(s_{t+1} | s_t, a_t)$ – функція переходу, ймовірність потрапити у стан s_{t+1} після виконання дії a_t у стані s_t .

$R(s_t, s_{t+1})$ – нагорода, отримана за перехід зі стану s_t у s_{t+1} .

γ – коефіцієнт дисконтування $0 < \gamma < 1$, який зменшує вагу майбутніх нагород.

Навчання з підкріпленням є особливо ефективним для вирішення складних завдань, зокрема таких, як клінічне прийняття рішень, оскільки RL-агенти здатні приймати дії навіть у процесі незавершеного навчання та послідовно коригувати свої стратегії для максимізації очікуваної винагороди. Хоча алгоритми навчання під наглядом також можуть виконувати завдання з неповних даних шляхом імпутації, такі заміни не завжди точні та можуть призвести до прийняття неправильних рішень [43, 44].

Оскільки винагороди в RL можуть бути рідкісними та відстроченими, такі алгоритми здатні відстежувати зв'язок між дією та відкладеним результатом, що є складним завданням для традиційних підходів [30]. Це дає змогу RL-агентам ефективно розв'язувати задачі, у яких наслідки рішень проявляються з часом.

У складних клінічних завданнях, наприклад, при онкологічних захворюваннях, RL демонструє значний потенціал. Завдяки здатності обробляти великі масиви даних, зокрема візуалізацій, біомедичних показників та індивідуальних характеристик пацієнта, RL може допомогти оптимізувати стратегії планування та моніторингу лікування в реальному часі. Ця властивість є критично важливою в онкології, де лікування має бути адаптивним і персоналізованим, а відгук на терапію може бути непередбачуваним. Крім того, RL здатне враховувати довготривалі наслідки клінічних рішень, навіть за умов неповного розуміння зв'язків між терапевтичними діями та кінцевими результатами.

Варто підкреслити, що RL не належить ні до контрольованого, ні до неконтрольованого навчання. На відміну від перших, де моделі вивчаються на основі мічених прикладів, і других, які намагаються виявити приховані закономірності в неструктурованих даних, RL ґрунтується на взаємодії агента із середовищем, де зворотний зв'язок реалізується через систему винагород та штрафів [47]. Такий підхід дозволяє агенту адаптувати свою політику, орієнтуючись на реальні наслідки своїх дій.

Унікальною проблемою RL є необхідність балансу між експлуатацією (вибір добре відомих дій, що приносять винагороду) та дослідженням (випробування нових дій для можливого покращення стратегії). Надмірна увага до лише одного з цих аспектів веде до неоптимальних результатів, тому цей компроміс є предметом інтенсивного дослідження [45]. Подібна дилема відсутня у класичних задачах контрольованого або неконтрольованого навчання, що робить RL концептуально відмінним і більш гнучким у деяких сферах.

RL-методи можна поділити на модельні (model-based) та безмодельні (model-free). У модельному RL агент будує уявлення про динаміку середовища — наприклад, у симульованих середовищах з відомими правилами — і на основі цієї моделі планує дії. У безмодельному підході

агент не має доступу до інформації про перехідні й винагородні функції середовища і вчиться виключно з досвіду за допомогою стратегій проб і помилок [42].

Безмодельні алгоритми діляться на моделі, які засновані на цінностях (Value-based RL), засновані на політиці (Policy-based RL) та акторсько-критичні методи (Actor-critic RL).

Метою моделей, які засновані на цінностях, є вивчення цінності станів або ціннісних функцій для отримання оптимальної політики прийняття рішень. Один з найбільш поширених алгоритмів навчання з підкріпленням Q-learning дозволяє агенту знаходити оптимальну стратегію дій у динамічному середовищі шляхом максимізації суми майбутніх винагород.

Методи, що засновані на політиці, - це методи градієнта політики, які діють шляхом безпосередньої оптимізації очікуваної винагороди зі знижкою для отримання оптимальної політики. Типовий метод цього класу алгоритмів - State–Action–Reward–State–Action (SARSA) – використовує марковський процес прийняття рішень (МПП), який є стохастичною задачею керування для послідовного прийняття рішень, коли результати невизначені.

Акторсько-критичні методи поєднують ідею градієнта політики та підходів, що засновані на цінностях, для подолання відповідних обмежень. Градієнтні методи політики часто збігаються до локальних максимумів і мають тенденцію до високої дисперсії, тоді як методи, що засновані на значеннях, можуть мати слабку збіжність. Алгоритми актора-критика вдосконалюють ці проблеми, використовуючи актора для визначення дій на основі градієнтів політики, а критика для оцінки якості дій за допомогою навчання часових відмінностей сприяє ефективному оновленню в безперервних просторах дій. Прикладами є методи A2C, A3C [45].

Таким чином, на сучасному етапі різні методи і алгоритми ML використовуються для обробки даних, розпізнавання, класифікації, прогнозування і прийняття рішень у складних багатопараметричних задачах практичної онкології з невизначеними розв'язками. У зв'язку з важливістю і складністю математичних підходів і алгоритмів в останні роки сформувалась математична онкологія - міждисциплінарна галузь, яка використовує сучасні математичні та обчислювальні методи для вивчення початку, прогресування та реакції на лікування раку. Математична онкологія поєднує математику, інформатику та біологію для створення моделей, що описують рак на різних рівнях, від молекулярного (генетичні показники та інші –оміки) до клітинного і тканинного (зростання пухлини та поширення метастаз).

2. Результати і висновки о перспективах використання машинного навчання в онкології

В сучасній онкології вибір метода машинного навчання, який буде використовуватись, залежить від задачі, яку потрібно буде вирішити. Переваги та недоліки різних методів, які використовувалися у сучасних дослідженнях, наведені в Таблиці 1.

Таблиця 1. Порівняння використання методів машинного навчання в онкології
Table 1. Comparison of the use of machine learning methods in oncology

Метод навчання	Приклади алгоритмів	Переваги	Недоліки	Завдання де метод найкраще використовується
Контрольоване навчання	Linear Models [50] [51] [53]	- Простота, швидкість - Висока інтерпретованість результатів	- Потребує мічених даних, не працює добре для нових мутацій - Не виявляє складні нелінійні зв'язки - Погано працюють з високорівневими залежностями	Прогноз виживаності, ризику, раннє виявлення
	Decision Tree Models [57]	- Інтерпретовані правила - Працює з категоріальними та числовими даними	- Чутливі до шуму - Можливість переобучення (overfitting)	Прогноз реакції на препарат, класифікація
	Support Vector Machine [58]	Добре працює з невеликими наборами, стійкий до переобучення	Важко масштабувати на великі дані	Класифікація пухлин, прогноз рецидивів

	Ensemble Models [52]	- Висока точність - Стійкість до переобучення	- Важко інтерпретувати - Довгий час тренування	Класифікація типу пухлини, прогноз відповіді на лікування
	Neural Networks [54] [55], [56]	- Модель нелінійних зв'язків, потужні для зображень - Добре масштабуються	- Потребують великого обсягу даних - Складність у налаштуванні, "чорна скринька"	Класифікація гістообразень, виявлення пухлин на сканах
Неконтрольоване навчання	K-means [39]	- Виявлення нових закономірностей - Простота реалізації - Швидка обробка великих наборів	- Важко інтерпретувати - Не завжди має клінічну цінність - Чутливість до вибору k - Не гарантує оптимального розбиття	- Виявлення нових підтипів раку - Стратифікація пацієнтів
	Autoencoders [59]	- Виявлення аномалій - Зниження розмірності	- Потребують великої кількості даних - Важко інтерпретувати	- Зниження розмірності даних - Виявлення рідкісних мутацій
Навчання з підкріпленням	SARSA [62]	- Оптимізація схем лікування - Ураховує реальну поведінку агента - Стабільніша поведінка	- Висока обчислювальна складність, складність валідації та інтерпретації - Повільніше навчання - Залежить від поточної політики	Оптимізація дозування, адаптивні протоколи лікування Інтерактивні рекомендації в персоналізованій терапії
	Q-learning [32][48][64][65]	- Швидше знаходить оптимальну політику - Підходить для складних середовищ	- Складність валідації та інтерпретації - Може бути нестабільним - Вимагає добре підбраної грейдентної стратегії	Оптимізація дозування, вибір послідовності лікування
	Reinforce [61]	- Простий у реалізації - Добре працює з безперервними діями	- Складність валідації та інтерпретації - Висока дисперсія градієнтів - Повільна збіжність	Навчання вибору тактик терапії (в контексті часу / відповіді)
	AC (Actor-Critic) [49] [63]	- Стабільніший і швидший за REINFORCE - Менша дисперсія оцінки	- Складність валідації та інтерпретації - Складніша реалізація - Чутливий до налаштувань гіперпараметрів	Динамічне управління протоколом лікування в реальному часі

Таким чином, обговорені у попередньому розділі методи контрольованого SL [50-58] та неконтрольованого UL [39,59] машинного навчання широко використовуються у медичних дослідженнях, переважно на основі ретроспективних даних. Їх успішно застосовують для вирішення таких завдань, як діагностика, стратифікація ризику, класифікація геномних підтипів, прогнозування результатів лікування, моніторинг рецидивів та передбачення відповіді пацієнта на терапію [60]. Ці підходи дозволяють отримувати важливі прогностичні моделі, зокрема в математичній онкології.

Однак реальна клінічна практика є надзвичайно динамічною: змінюються як стан пацієнта, так і зовнішні умови. Тому виникає питання, наскільки ефективно моделі, які були навчені на статичних ретроспективних даних, здатні працювати в умовах реальної динаміки за умовами невизначеності. В більшості випадків ці моделі не враховують динамічні зміни, які відбуваються під час лікування.

У цьому контексті навчання з підкріпленням RL постає як перспективна альтернатива. На відміну від SL та UL, RL має послідовну структуру прийняття рішень у часі, що дозволяє моделі адаптуватися до змін середовища, зокрема клінічного. RL-алгоритми здатні навчатися обирати найкращі дії в залежності від поточного стану пацієнта, з метою максимізації довгострокових клінічних результатів [42].

Цей підхід також узгоджується з принципами прецизійної медицини, яка орієнтується на індивідуалізацію лікування. Основна мета - підібрати терапію та дозування для кожного пацієнта з урахуванням його біологічних характеристик і особливостей перебігу захворювання, що не тільки збільшує шанси пацієнтів на виживання, але й покращує якість їхнього життя за рахунок зменшення небажаних побічних ефектів [37].

Такий підхід дозволяє підвищити ефективність лікування, зменшити ймовірність рецидивів і побічних ефектів, та адаптувати терапевтичні стратегії під конкретного пацієнта або групи пацієнтів, які найбільше виграють від інтервенції [42].

Методи навчання з підкріпленням показують кращі результати в порівнянні з іншими алгоритмами навчання, які використовувались для перевірки та валідації, в питаннях оптимізування та динамічного керування системами лікування. Так, дослідження [49], яке використовувало алгоритм Actor-Critic та акцентувало увагу на зв'язку між геномними особливостями та реакцією на ліки при лікуванні раку молочної залози, перевершило результати, які були отримані від інших п'яти методів: еластичної мережі ElasticNet (EN), регресія ядра-гребня (Kernel Ridge Regression, KRR), факторизація матриці подібності з регуляризациєю Similarity-Regularized Matrix Factorization (SRMF), прогнозування відповіді на препарати проти раку з використанням системи рекомендацій Cancer Drug Response prediction using a Recommender System (CaDRRes), та навчання з ранжуванням на основі ядра Kernelized Ranking Learning (KRL). Дослідження [61], яке використовує алгоритм Reinforce для ультразвукового дослідження розрізнення доброякісних та злоякісних вузлів молочної залози, показало гарні результати в порівнянні з використанням одноmodalних моделей та різних популярних мультимодальних підходів, включаючи Share-Net, Centralnet та Voting Schemes. В дослідженні [32] при плануванні лікування променевою терапією з модульованою інтенсивністю, Q-learning алгоритм показав результати не гірші ніж результати які були отримані лікарем, який керував системою планування лікування для коригування параметрів лікування, наприклад, об'єму дози, щоб досягти задовільного плану для кожного пацієнта.

Інші дослідження [62-65], які використовували навчання з підкріпленням, теж показали гарні результати в порівнянні з алгоритмами інших методів навчання.

3. Висновки

В роботі наведений детальний огляд сучасних методів розпізнавання, класифікації, прогнозування і обґрунтованого прийняття рішень відносно вибору хіміотерапії і усунення можливих шкідливих впливів (ADR) на здоров'я пацієнта з онкологічним захворюванням. Окрім методів математичної статистики, у математичній онкології широко використовуються методи глибокого навчання. Особливо ефективними є методи випадкових лісів (Random Forest), опорних векторних машин (SVM) і штучних нейронних мереж (ANN), які можуть аналізувати складні залежності між великою кількістю різноманітних параметрів від генетичних маркерів і хімічних речовин (ліків) до тривимірних КТ-зображень пухлини, яка зростає. Найбільш детально розглянуті методи контрольованого машинного навчання SL, неконтрольованого навчання UL та навчання з підкріпленням RL. У наведеній Таблиці 1 для кожного методу навчання наведені переваги, недоліки та ті завдання, де цей метод найкраще використовується. Показано, що найбільш перспективним є метод RL, який відкриває нові можливості для створення комп'ютерних систем підтримки клінічних рішень, що здатні динамічно коригувати лікування в реальному часі у відповідь до змін (позитивних або негативних) у стані пацієнта, а також відповіді організму на хіміотерапію та можливість виникнення небажаних побічних ефектів від використаної комбінації ліків. Отримані результати будуть використані у подальшій роботі.

REFERENCES

1. F. Bray, M. Laversanne, H. Sung, J. Ferlay, R.L. Siegel, I. Soerjomataram, A. Jemal, “Global cancer statistics 2022: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries”, *Cancer Journal for Clinicians*, 74:229–263, 2024. DOI: 10.3322/caac.21834
2. Global cancer observatory: website. URL: <https://gco.iarc.who.int> (accessed 12.05.2025)
3. W.D. Hazelton, E. G. Luebeck, “Biomarker-based early cancer detection: Is it achievable?”, *Sci Transl Med*;3(109):109fs9, 2011. DOI: 10.1126/scitranslmed.300327
4. C.D. de los Rios de la Serna, C.B. Boers-Doets, T. Wiseman, B. Radia, R. Hammond, “Early Recognition and Management of Side Effects Related to Systemic Anticancer Therapy for Advanced Breast Cancer”, *Seminars in Oncology Nursing*, 40, 151553, 2024. <https://doi.org/10.1016/j.soncn.2023.151553>
5. J.A. DiMasi, H.G. Grabowski, R.W. Hansen, “Innovation in the pharmaceutical industry: new estimates of R&D costs”, *Journal Health Econ*, 47 20–33, 2016. doi: 10.1016/j.jhealeco.2016.01.012
6. M. Nilashi, H. Ahmadi, R.A. Abumalloh, M. Alrizq, A. Alghamdi, S. Alyami, “Knowledge Discovery of Patients Reviews on Breast Cancer Drugs: Segmentation of Side Effects Using Machine Learning Techniques Application of artificial intelligence and machine learning in early detection of adverse drug reactions (ADRs) and drug-induced toxicity”, *Artificial Intelligence Chemistry* 1, 100011, 2023. <https://doi.org/10.1016/j.aichem.2023.100011>
7. S. Yang, S. Kar. Application of artificial intelligence and machine learning in early detection of adverse drug reactions (ADRs) and drug-induced toxicity, 2023, USA <https://doi.org/10.1016/j.aichem.2023.100011>
8. R. P. Riechelmann, C. Zimmermann, S. N. Chin, L. Wang, A. O’Carroll, S. Zarinehbafe, “Potential drug interactions in cancer patients receiving supportive care exclusively”, *Journal of Pain and Symptom Management*, 35(5), 535–543, 2008. doi: 10.1016/j.jpainsymman.2007.06.009
9. J. Sultana, P. Cutroneo, G. Trifiro, “Clinical and economic burden of adverse drug reactions”, *Journal Pharmacol Pharmacother*, 4, 73–77, 2013. doi: 10.4103/0976-500X.120957
10. A. Bhandari, B. Gu, F.M. Kashkooli, W. “Zhan Image-based predictive modelling frameworks for personalised drug delivery in cancer therapy” *Journal of Controlled Release*, 370, 721–746, 2024. <https://doi.org/10.1016/j.jconrel.2024.05.004>
11. R. C. Rockne, J. G. Scott, “Introduction to Mathematical Oncology” *JCO Clin Cancer Inform*. 3:CCI.19.00357, Apr 26, 2019. doi: 10.1200/CCI.19.00010
12. F. Firoozbakht, B. Yousefi, B. Schwikowski, “An overview of machine learning methods for monotherapy drug response prediction”, *Briefings in Bioinformatics*, pp.bb408, 2021. doi: 10.1093/bib/bbab408. hal-03376028
13. O. Teplytska, M. Ernst, L.M. Koltermann, D. Valderrama, E. Trunz, M. Vaisband, J. Hasenauer, H. Fröhlich, H. Jaehde Machine, “Learning Methods for Precision Dosing in Anticancer Drug Therapy: A Scoping Review”, *Clinical Pharmacokinetics* 63:1221–1237, 2024. doi:10.1007/s40262-024-01409-9
14. T. Eguale, D.L. Buckeridge, A. Verma, N.E. Winslade, A. Benedetti, J.A. Hanley, “Association of off-label drug use and adverse drug events in an adult population”, *JAMA Internal Medicine*, 176(1), 55–63, 2016. doi: 10.1001/jamainternmed.2015.6058.
15. S. Guan, G. Wang, “Drug discovery and development in the era of artificial intelligence: From machine learning to large language models”, *Artificial Intelligence Chemistry*, 2, 100070, 2024. <https://doi.org/10.1016/j.aichem.2024.100070>
16. J. Liu, M. Li, X. Chen, “AntiMF: A deep learning framework for predicting anticancer peptides based on multi-view feature extraction”, *Methods*, 207, 38–43, 2022. <https://doi.org/10.1016/j.ymeth.2022.07.017>
17. L.T. Phan, H.W. Park, T. Pitti, T. Madhavan, Y.-J. Jeon, B. Manavalan, “MLACP 2.0: An updated machine learning tool for anticancer peptide prediction”, *Computational and Structural Biotechnology Journal*, 20, 4473–4480, 2022. doi:10.1016/j.csbj.2022.07.043
18. B.A. Badwan, G. Liaropoulos, E. Kyrodimos, D. Skaltsas, A. Tsigirigos, V.G. Gorgoulis, “Machine learning approaches to predict drug efficacy and toxicity in oncology”, *Reports Methods*, 3, 100413, 2023. doi:10.1016/j.crmeth.2023.100413

19. A. Partin, T.S. Brettin, Y. Zhu, O. Narykov, A. Clyde, J. Overbeek and R.L. Stevens, “Deep learning methods for drug response prediction in cancer: Predominant and emerging trends”, *Front. Med.* 10:1086097, 2023. doi: 10.3389/fmed.2023.1086097
20. K. Fan, L. Cheng, L. Li, “Artificial intelligence and machine learning methods in predicting anti-cancer drug combination effects”, *Briefings in Bioinformatics*, 22(6), 1–12, 2021. <https://doi.org/10.1093/bib/bbab271>
21. S. She, H. Chen, W. Ji, M. Sun, J. Cheng, C. RuiMand Feng, “Deep learning based multi-drug synergy prediction model for individually tailored anticancer therapies”, *Front. Pharmacol.* 13:1032875, 2022. doi: 10.3389/fphar.2022.1032875
22. R. Rafique, S.M.R. Islam, J.U. Kazi, “Machine learning in the prediction of cancer therapy”, *Computational and Structural Biotechnology Journal*, 19, 4003–4017, 2021. doi.org/10.1016/j.csbj.2021.07.003
23. J.-B. Zhou, D. Tang, L. He, S. Lin, J.H. Lei, H. Sun, X. Xu, C.-X. Deng, “Machine learning model for anti-cancer drug combinations: Analysis, prediction, and validation”, *Pharmacological Research*, 194, 106830, 2023. doi:10.1016/j.phrs.2023.106830
24. W. Shan, C. Shen, L. Luo, P. Ding, “Multi-task learning for predicting synergistic drug combinations based on auto-encoding multirelational graphs”, *iScience*, 26, 108020, 2023. doi.org:10.1016/j.isci.2023.108020
25. M. Timilsina, M. Tandan, V. Nováček. “Machine learning approaches for predicting the onset time of the adverse drug events in oncology”, *Machine Learning with Applications*, 9, 100367, 2022. doi:10.1016/j.mlwa.2022.100367
26. M.S. Barakat, M. Field, A. Ghose, D. Stirling, L. Holloway, S. Vinod, “The effect of imputing missing clinical attribute values on training lung cancer survival prediction model performance”, *Health Information Science and Systems*, 5(1), 1–11, 2017. DOI:10.1007/s13755-017-0039-4
27. R. Kurilov, B. Haibe-Kains, B. Brors, “Assessment of modelling strategies for drug response prediction in cell lines and xenografts”, *Sci Rep*, 10:2849, 2020. DOI:10.1038/s41598-020-59656-2
28. T. Hastie, R. Tibshirani, J. Friedman, *The elements of statistical learning data mining, inference, and prediction*, New York, NY: Springer Science & Business Media, 2009.
29. Y. Li, D.M. Umbach, J.M. Krahn, “Predicting tumor response to drugs based on gene-expression biomarkers of sensitivity learned from cancer cell lines”, *BMC Genomics*, 22:272, 2021. DOI:10.1186/s12864-021-07581-7
30. S. Nath, E. Korot, D.J. Fu, G. Zhang, K. Mishra, A.Y. Lee, “Reinforcement learning in ophthalmology: potential applications and challenges to implementation”, 4, E692-E697, September, 2022. DOI: 10.1016/S2589-7500(22)00128-5
31. A. Jonsson, “Deep reinforcement learning in medicine”, *Kidney Dis. (Basel)*, 5(1):18–22, 2019. doi: 10.1159/000492670
32. C. Shen, D. Nguyen, L. Chen, Y. Gonzalez, R. McBeth, N. Qin, S. B. Jiang, and X. Jia, “Operating a treatment planning system using a deep-reinforcement learning-based virtual treatment planner for prostate cancer intensity-modulated radiation therapy treatment planning,” *Medical physics*, vol. 47, no. 6, pp. 2329–2336, 2020. DOI: 10.1002/mp.14114
33. I. Triguero, S. García, F.Herrera, “Self-labeled techniques for semi-supervised learning: taxonomy, software and empirical study”, *Knowledge and Information Systems*, 42 (2):245–84, 2015. DOI:10.1007/s10115-013-0706-y
34. C. S. Greene, J. Tan, M. Ung, J. H. Moore, C. Cheng, “Big Data Bioinformatics”, *J Cell Physiol*, 229(12):1896-900, Dec. 2014. DOI: 10.1002/jcp.24662
35. C. Cortes, V. Vapnik, “Support-vector networks”, *Chemical Biology & Drug Design*, 297(3):273-297, 2009. DOI:10.1007/%2FBF00994018
36. R.Tibshirani, “Regression shrinkage and selection via the Lasso”, *Journal of the Royal Statistical Society Series B (Statistical Methodology)*, 73(3):273-282, 2011. DOI:10.2307/41262671
37. R. Rafique, S.M. Riazul Islam, J. U. Kazi, “Machine learning in the prediction of cancer therapy”, *Comput Struct Biotechnol J*, 8:19:4003-4017, Jul., 2021. DOI: 10.1016/j.csbj.2021.07.003
38. V. Podgorelec, P. Kokol, B. Stiglic, I.Rozman, “Decision trees: an overview and their use in medicine”, *J. Med. Syst.*, 26:445–63, 2002. doi: 10.1023/a:1016409317640.
39. R.W. Tothill, A.V. Tinker, J. George, R. Brown, S.B. Fox, S. Lade, D.S. Johnson, M.K. Trivett, D. Etemadmoghadam, B. Locandro, N. Traficante, S. Fereday, J.A. Hung, Y.E. Chiew, I. Haviv, D. Gertig, A. DeFazio, D.D. Bowtell, “Novel molecular subtypes of serous and endometrioid ovarian

- cancer linked to clinical outcome”, *Clin Cancer Res.*, 14:5198–5208, 2008. doi: 10.1158/1078-0432.CCR-08-0196.
40. L. Breiman, “Random forests”, *Machine Learning*, 45:5–32, 2001. DOI:10.1023/A:1010950718922
 41. D. Bertsimas, H. Wiberg, “Machine Learning in Oncology: Methods, Applications, and Challenges”, *JCO Clin Cancer Inform*, 4:885-894, Oct. 2020. doi: 10.1200/CCI.20.00072.
 42. J.-N. Eckardt, K. Wendt, M. Bornhäuser, J. M. Middeke, “Reinforcement Learning for Precision Oncology”, *Cancers (Basel)*, 13(18):4624, Sep. 2021. doi: 10.3390/cancers13184624.
 43. S. Van Buuren, *Flexible imputation of missing data*, CRC. Press, New York, NY, 2018
 44. D. Pratap, *Statistics for Machine Learning Build supervised, unsupervised, and reinforcement learning models using both Python and R.*, Birmingham, England, Packt, 2017.
 45. R.G. Lopez, “Reinforcement Learning in Oncology: A Comprehensive Review”. 2023 <https://studenttheses.uu.nl/handle/20.500.12932/47144>
 46. V. Mnih, K. Kavukcuoglu, D. Silver, “Human-level control through deep reinforcement learning”, *Nature*, 518(7540):529-33, Feb, 2015 DOI: 10.1038/nature14236
 47. R. S. Sutton, A. Barto, *Reinforcement learning: An introduction*, MIT press, London, England, 2018.
 48. C. Li, Y. Guo, X. Lin, X. Feng, D. Xu, R. Yang, “Deep reinforcement learning in radiation therapy planning optimization: A comprehensive review”, *Phys Med*, 125:104498, Sep, 2024. doi: 10.1016/j.ejmp.2024.104498.
 49. M. Liu, X. Shen, W. Pan, “Deep reinforcement learning for personalized treatment recommendation”, *Stat Med*, 41(20):4034-4056, Sep, 2022. doi: 10.1002/sim.9491.
 50. J.S. Parker, M. Mullins, M.C.U. Cheang, S. Leung, D. Voduc, T. Vickery, “Supervised risk predictor of breast cancer based on intrinsic subtypes”, *J. Clin. Oncol*, 27(8):1160–7, 2009. doi: 10.1200/JCO.2008.18.1370
 51. P. Geeleher, N.J. Cox, R. Huang, “Clinical drug response can be predicted using baseline gene expression levels and in vitro drug sensitivity in cell lines”, *Genome Biol*, 15(3):R47, 2014. doi: 10.1186/gb-2014-15-3-r47.
 52. Y. Kim, D. Kim, B. Cao, R. Carvajal, M. Kim, “PDXGEM: patient-derived tumor xenograft-based gene expression model for predicting clinical response to anticancer therapy in cancer patients”, *BMC Bioinformatics*, 21(1):288, Jul, 2020. doi: 10.1186/s12859-020-03633-z.
 53. J.H. Kong, H. Lee, D. Kim, S.K. Han, D. Ha, K. Shin, “Network-based machine learning in colorectal and bladder organoid models predicts anti-cancer drug efficacy in patients”, *Nat Commun*, 11(1):5485, Oct 2020. doi: 10.1038/s41467-020-19313-8.
 54. M.P. Menden, F. Iorio, M. Garnett, U. McDermott, C.H. Benes, P.J. Ballester, “Machine learning prediction of cancer cell sensitivity to drugs based on genomic and chemical properties”, *PLoS One*, 8(4):e61318, Apr 2013. doi: 10.1371/journal.pone.0061318. Print 2013.
 55. Y. Chang, H. Park, H.J. Yang, S. Lee, K.Y. Lee, T.S. Kim, “Cancer Drug Response Profile scan (CDRscan): A Deep Learning Model That Predicts Drug Effectiveness from Cancer Genomic Signature”, *Sci Rep*, 8(1):8857, Jun 2018. doi: 10.1038/s41598-018-27214-6.
 56. T. Sakellaropoulos, K. Vougas, S. Narang, F. Koinis, A. Kotsinas, A. Polyzos, “A Deep Learning Framework for Predicting Response to Therapy in Cancer”, *Cell Rep*, 29(11):3367-3373.e4, Dec, 2019. doi: 10.1016/j.celrep.2019.11.017..
 57. A. Malyutina, M.M. Majumder, W. Wang, A. Pessia, C.A. Heckman, J. Tang, “Drug combination sensitivity scoring facilitates the discovery of synergistic and efficacious drug combinations in cancer”, *PLoS Comput Biol*, 15(5):e1006752, May, 2019. doi: 10.1371/journal.pcbi.1006752
 58. M. Ammad-Ud-Din, S.A. Khan, K. Wennerberg, T. Aittokallio, “Systematic identification of feature combinations for predicting drug response with Bayesian multi-view multi-task linear regression”, *Bioinformatics*, 33(14):i359-i368, Jul 2017. doi: 10.1093/bioinformatics/btx266.
 59. T. Zhang, L. Zhang, P.R.O. Payne, F. Li, “Synergistic Drug Combination Prediction by Integrating Multiomics Data in Deep Learning Models”, *Methods Mol Biol*, 2194:223-238, 2021. doi: 10.1007/978-1-0716-0849-4_12.
 60. K.-H. Yu, A.L. Beam, I.S. Kohane, “Artificial Intelligence in Healthcare”, *Nat Biomed Eng*, 2(10):719-731, Oct, 2018. doi: 10.1038/s41551-018-0305-z.
 61. J. Wang, J. Miao, X. Yang, R. Li, G. Zhou, Y. Huang, Z. Lin, W. Xue, X. Jia, J. Zhou, “Auto-weighting for breast cancer classification in multimodal ultrasound”, *Springer*, Part VI 23, pp. 190–199, 2020. DOI:10.48550/arXiv.2008.03435

62. F. E. Alsaadi, A. Yasami, C. Volos, S. Bekiros, and H. Jahanshahi, "A new fuzzy reinforcement learning method for effective chemotherapy", *Mathematics*, vol. 11, no. 2, p. 477, 2023. DOI:10.3390/math11020477
63. C. Xu, Y. Song, D. Zhang, L. K. Bittencourt, S. H. Tirumani, and S. Li, "Spatiotemporal knowledge teacher–student reinforcement learning to detect liver tumors without contrast agents", *Medical Image Analysis*, vol. 90, p. 102 980, 2023. doi: 10.1016/j.media.2023.102980.
64. C. Shen, D. Nguyen, L. Chen, Y. Gonzalez, R. McBeth, N. Qin, S. B. Jiang, and X. Jia, "Operating a treatment planning system using a deep-reinforcement learning-based virtual treatment planner for prostate cancer intensity-modulated radiation therapy treatment planning", *Medical physics*, vol. 47, no. 6, pp. 2329–2336, 2020. doi: 10.1002/mp.14114.
65. A. Ebrahimi Zade, S. Shahabi Haghighi, M. Soltani, "Reinforcement Learning for Optimal Scheduling of Glioblastoma Treatment with Temozolomide", *Comput Methods Programs Biomed*, 193:105443, Sep 2020. doi: 10.1016/j.cmpb.2020.105443

Tiurdo Ivan *PhD of the Department of Applied Mathematics, V. N. Karazin Kharkiv National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine;*

Kizilova Natalya *Doctor of Physical and Mathematics Sciences, Professor; professor of the Department of Applied Mathematics, V. N. Karazin Kharkiv National University, 4 Svobody Sq., Kharkiv, 61022, Ukraine;*

The use of machine learning methods in modern mathematical oncology

The purpose of the study is to analyze modern approaches to assessing the efficacy and safety of anti-cancer drugs using machine learning methods, as well as to determine the prospects for their use in modern mathematical oncology, in oncological research that widely uses mathematical modelling and simulation.

As a result of the study, a literature review was conducted on the use of machine learning in oncology. An analysis of the use of the main machine learning methods, such as Supervised Learning (SL), Unsupervised Learning (UL), and **Reinforcement Learning (RL)** in modern oncology was performed. Examples of the use of various machine learning algorithms in research related to anti-cancer therapy and oncology in general are given. The advantages and disadvantages of the used machine learning algorithms are analyzed depending on the tasks to be solved.

Conclusions: Machine learning methods are already widely used in medical research in the field of oncology. They have been successfully applied to solve many issues and showed good results, but there are still many areas in oncology where the use of machine learning methods can make a significant contribution to improving medical research and medical care in the treatment of cancer. For example, the use of algorithms based on reinforcement learning in precision medicine looks very promising, as its methods play a significant role in the personalized treatment of cancer.

Keywords: *machine learning, supervised learning, unsupervised learning, reinforcement learning, oncology, mathematical oncology, precision medicine, individual medicine, medical research.*

УДК (UDC) 004.891

Shabelnyk Tetiana

*Doctor of Economic Sciences, Professor;
Head of the Department of Economic Cybernetics and System Analysis,
Simon Kuznets Kharkiv National University of Economics , ave. Sciences,
9-A, Kharkiv, 61166, Ukraine;
e-mail: Tanya.shabelnik17@gmail.com
<https://orcid.org/0000-0001-9798-391X>*

Prokopovych Svitlana

*Candidate of Economic Sciences, Associate Professor;
Associate Professor of the Department of Economic Cybernetics and
System Analysis,
Simon Kuznets Kharkiv National University of Economics , ave. Sciences,
9-A, Kharkiv, 61166, Ukraine
e-mail: prokopovichsv@gmail.com ;
<https://orcid.org/0000-0002-6333-2139>*

An expert system for evaluating language patterns using nonparametric statistics

Abstract. The paper focuses on the research of language regularities by methods of non-parametric statistics. The emphasis is placed on how to resolve the common problems regarding the use of non-parametric statistical techniques, in particular the one of semantic marking of sentences in linguistics.

The aim of the scientific research is defined as the development of an expert system for evaluating language regularities using non-parametric statistical methods for any language.

Research methods. Methods of non-parametric statistics, IDEF4 notation, the programming language C #.

The research offers a specialized expert system that uses the C Programming Language to provide automatic questioning of the native speakers with further analysis of the lexical meaning of word combinations and phrases implemented as a Desktop Program for Windows. The program considers the possibility of taking into account of how to reveal the correspondence of phrases from dictionary files that have different expert assessments.

Keywords: *expert system, non-parametric statistical methods, language regularities, software.*

Як цитувати: Shabelnyk T., Prokopovych S. An expert system for evaluating language patterns using nonparametric statistics. *Вісник Харківського національного університету імені В. Н. Каразіна, серія Математичне моделювання. Інформаційні технології. Автоматизовані системи управління.* 2025. вип. 68. С.113-119. <https://doi.org/10.26565/2304-6201-2025-68-11>

How to quote: T. Shabelnyk., S. Prokopovych, “An expert system for evaluating language patterns using nonparametric statistics”, *Bulletin of V. N. Karazin Kharkiv National University, series Mathematical modelling. Information technology. Automated control systems*, vol. 68, pp. 113-119, 2025. <https://doi.org/10.26565/2304-6201-2025-68-11>

Introduction

It is a well-known fact that the structure of any language and its development come to operate under the guidance of statistical laws. Language, as a system of communication characterized by the combination of discrete units, is represented by a certain way of organization, its particular structure and elements interconnected by subsystems of relationships. A constant increase in their number proves to be the result of the consequential evolution and development of the language. The use of statistical methods in linguistics is sure to increase the efficiency of linguistic observation, its accuracy and reliability as well as consistency and provability of linguistic assumptions.

1. Analysis Of Recent Researches And Publication

Prominent Ukrainian and foreign theorists and scholars (B. Strickland [1], V. Zhukovska [2], V. Zayats and M. Zayats [3], Ye. V. Kupriianov, N. S. Uholnikova, and O. M. Yurchenko [4], A. Sitar [6]) published their original works dedicated to the issues of using statistical methods in linguistic research.

The large number of papers comes to indicate an increasing level of interest in the subject in question. Thus, advancing statistical methods in linguistics as well as practical application of statistical criteria claims to be more forward looking and based on the latest scientific evidence can lead to generate relevant issues for further research and investigation.

It should be noted, that most of the research done in the fields of linguistics is of a qualitative nature. normal distribution law used in linguistic research [4] is not always observed in the frequency distribution of language elements. Consequently, when the latter differs from the norm, other methods to study language patterns presume to be essential. In this regard, we consider it relevant to use nonparametric statistical methods based on empirical data, which, however, do not depend on their distribution law [4]. Succinctly, to provide real patterns of particular phenomena when there is a need to study inaccurate numerical values (qualitative), valid results come to be provided by using non-parametric criteria to identify differences in language elements.

As an illustration of the problem in question, the paper attempts to evoke the necessity of studying the issues of semantic distinctiveness of sentences. In performing these, a psycholinguistic experiment, or in other words, native speakers' perception is applied as a certain valuable technique to determine the significance (not materiality) with respect to experts' differences of opinion.

We also based on the research of Wen Zhang, Taketoshi Yoshida, [Xijin Tang](#) [7] where the authors asserts that a series of experiments on classifying the Reuters-21578 documents using the representations with multi-words. We used the performance of representation in individual words as the baseline, which has the largest dimension of feature set for representation without linguistic preprocessing. Moreover, linear kernel and non-linear polynomial kernel in support vector machines (SVM) are examined comparatively for classification to investigate the effect of kernel type on their performances. Index terms with low information gain (IG) are removed from the feature set at different percentages to observe the robustness of each classification method [7].

Lytvyn, V., Vysotska, V., Hrendus M. [8] mentioned who the main part of KB is ontology as clearly structured SA's model, systematic set of terms, which explain the connections between objects of this SA. Ontologies are generally accepted and widely used in different branches of science such as knowledge engineering, presentation of knowledge, information search, knowledge management, database design, information modeling and object-oriented analysis [8].

2. The Main Tasks Of The Study And Their Significance

Thus, our main objective is defined as the development of an expert system for evaluating language regularities using non-parametric statistical methods for any language. In its general formulation, the problem is represented in the way when two sentences come to be studied by experts who further evaluate their correct and precise usage in the language. Based on the results of their evaluation, a conclusion is made regarding the perception of the distinction between two sentences (either there is one or not).

These can be illustrated by the examples in Ukrainian: “Підприємець зробив розрахунки” / “Підприємець виконав розрахунки” and accordingly in English: “The Entrepreneur made calculations” / “The Entrepreneur performed calculations”.

To consider the example of expert analysis (Sentence 1 – The entrepreneur made calculations; Sentence 2 – The entrepreneur performed the calculations) one approach would be to demonstrate positive and negative assessments of sentences distinction made by 20 experts who are native speakers. The results of the evaluation are presented in Table 1.

Table 1. Expert Assessments of Sentence Distinction

Табл. 1. Експертні оцінки відмінності речень

Sentence	Expert assessment		
	Distinction available	Distinction nonavailable	Total
1. The entrepreneur made calculations	14 (a ₁)	6 (b ₁)	20
2. The entrepreneur performed the calculations	6 (a ₂)	14 (b ₂)	20

According to the Table 1, its first row is the inverse of the second. Consequently, the issues that come to reveal on how strong (weak) the difference in these rows look like have to be investigated using the criterion χ^2 [4] presented below:

$$\chi^2 = \frac{(a_1 - b_1 - 1)^2}{a_1 + b_1}, \quad (1)$$

where a_1 is the number of positive differences of the first sentence;

b_1 is the number of negative differences of the first sentence.

The result of appropriate calculations of criterion χ^2 according to the above example is represented as follows:

$$\chi^2 = \frac{(14 - 6 - 1)^2}{14 + 6} = 2,45. \quad (2)$$

In the next step, we determine the number of degrees of freedom that equals to:

$$f = (n - 1) \cdot (m - 1), \quad (3)$$

with n as the number of columns in the table;

whereas m is the number of rows in the table.

Thus we have the following calculation of degrees of freedom:

$$f = (2 - 1) \cdot (2 - 1) = 1. \quad (4)$$

Further on, we find the confidence limits of the criterion in the distribution table χ^2 with a reliability of 5% and 1%, thus, having values of 3.84 and 6.63, respectively. The calculated value of χ^2 , which is equal according to expression (2) 2.45 is lower than the tabular value. As a result, we conclude that the differences in the answers of experts are random and with a probability of 99% it can be resolved that in native speakers' minds these sentences are distinguished by certain difference.

3. Major Research Results

Forming a database of results of expert evaluations of lexically unambiguous phrases is a very painstaking and time-consuming process. We also note that there are difficulties in further analysis of large volumes of texts, finding correspondences in it from the database of phrases as well as the introduction of new phrases and their evaluation by experts that are native speakers. The rise of digital challenges that result in accelerated digitalization of information with its rapid growth and development, has a considerable impact on linguistics and science, in particular linguistic analysis of the text.

The latter, as known, with all its semantic unambiguity requires maximum automation. With this aim, the study has developed specific software. To carry out the automatic survey, an expert system has been developed by querying native speakers and further analyzing the lexical meaning of the phrase in accordance with mathematical expressions (1) – (4) which is implemented in the form of a desktop program.

To select the programming language of the expert system, we have formulated the following requirements: the software product must be implemented for Windows, provide a high-quality and friendly interface, and thus, there is no need for any distributed application.

At the present stage of software development tools, there have been many possible options to choose different resources to implement the task.

According to specialized ratings, the most popular programming languages are Python, Java, C #, C ++, each of them has a number of advantages and peculiar features. Python programming language is the option, which is convenient enough when one deals with the first language most frequently used for academic purposes. However, this language significantly loses in performance to C and Java, so it is not used to develop highly loaded applications and, consequently, to process large amounts of text.

The C # language is based on Java, only with a syntax closer to C ++ and is used mainly for the development of software products for the NET platform. Windows Framework [9]. Thus, the most effective way to solve this problem is to use the programming language C #. The Microsoft Visual Studio IDE has been developed especially for the C # language.

The algorithm of the program is presented in the form of an IDEF4 diagram (Fig. 1).

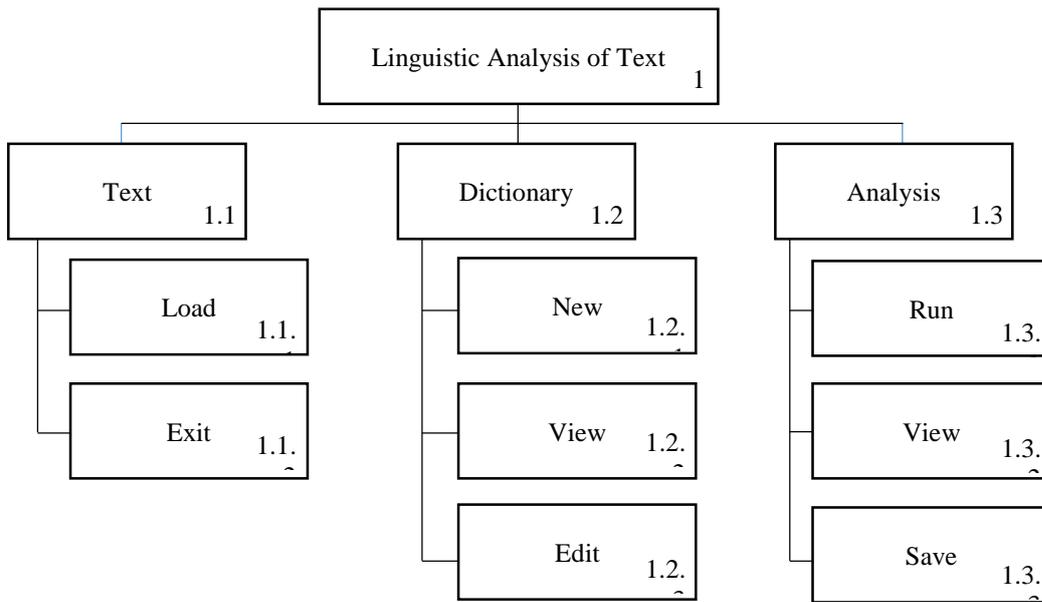


Fig. 1. Algorithmic Model of Expert System
 Рис.1. Алгоритмічна модель експертної системи

In the linguistic analysis of the text, there are three main actions – reading the text, compiling dictionaries of phrases with their expert assessments and directly assessing the text for linguistic homogeneity. The software does not provide the ability to edit the source document. The program reads text in docx, rtf or txt formats.

The algorithm provides the ability to create a new dictionary of phrases with the introduction of their expert assessments, viewing and downloading existing dictionaries.

Fig. 2 represents information model of the program. The text file and the dictionary file are loaded into RAM in a dialog mode. Depending on the search results of phrases in the text, the criterion χ^2 and the degree of freedom f are calculated.

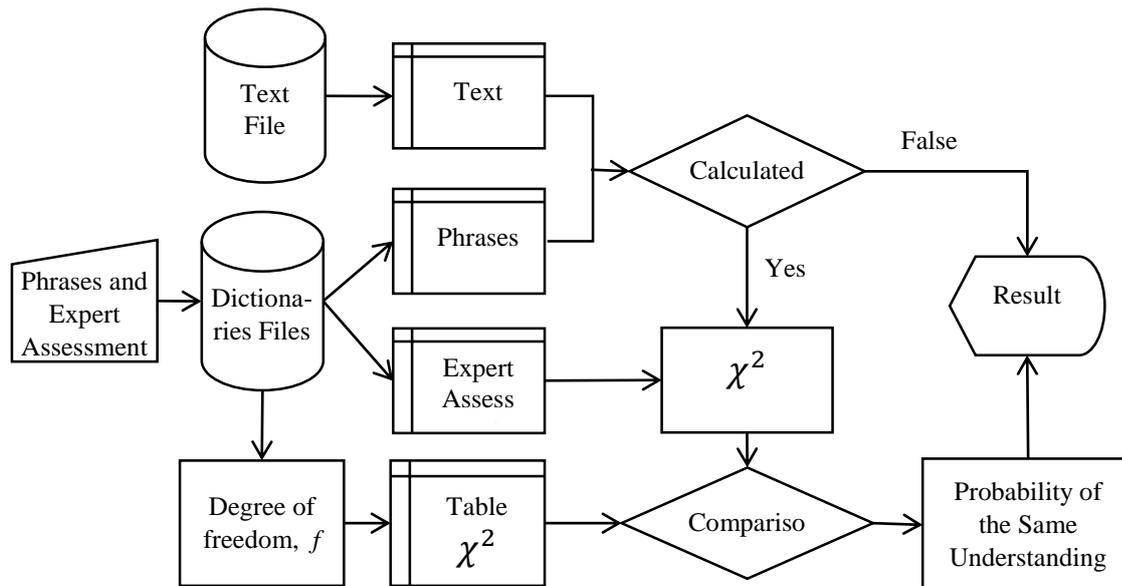


Fig. 2. Information Model of the Expert System
 Рис.2. Інформаційна модель експертної системи

When analyzing the text, it is possible to match several phrases from dictionary files with different expert assessments. In this case, the total probability of the same understanding of the text is calculated as the product of the probabilities of understanding each separate phrase:

$$P_{com} = P_1 \cdot P_2 \cdot \dots \cdot P_n, \quad (5)$$

where n is the number of identical phrases found.

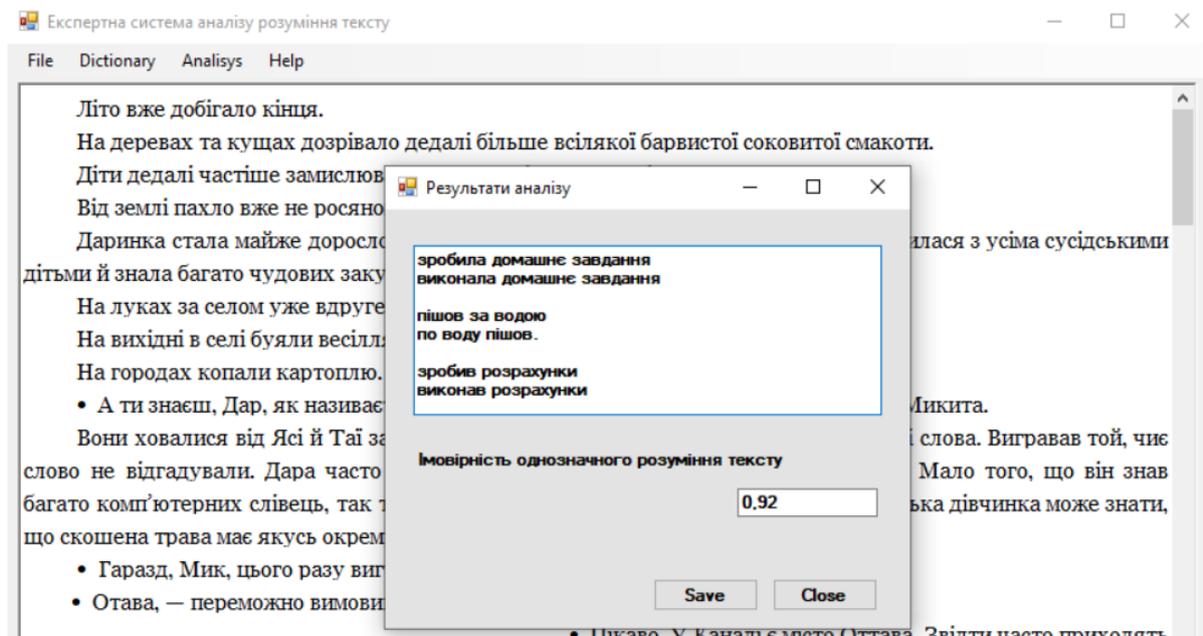


Fig. 3. The Results of Text Analysis Using an Expert System

Рис.3. Результати аналізу тексту з використанням експертної системи

The probability of unambiguous understanding of the text was 0.92. This means that in the minds of native speakers the sentences under consideration are not distinguished by difference.

Fig. 4 represents the results of studies of texts with different frequencies v coincidences of identical phrases are given.

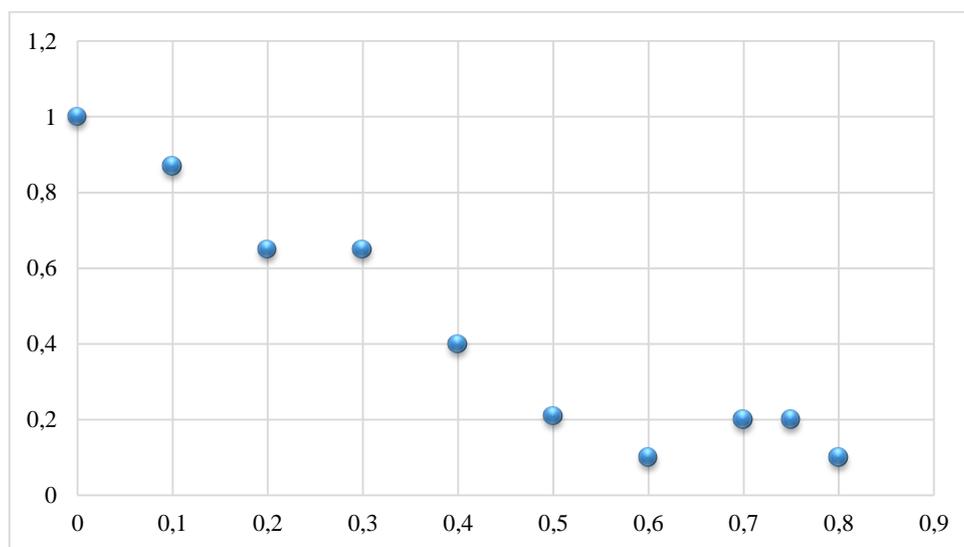


Fig. 4. f graphics

Рис.4. f графік

With sufficient accuracy, the probability of unambiguous understanding of the text P can be described by the function:

$$P = e^{-2,644 \cdot v}, R^2 = 0,8535 \quad (6)$$

The frequency of matching phrases from the dictionary is calculated as the ratio of the number of k matching phrases from the dictionary with different probabilities of understanding to the total number K of sentences in the text.

4. Conclusion

In linguistic research of qualitative characteristics of texts, nonparametric criteria of differences of language elements prove to provide more precise and rigorous results. In the context of smart integration of digital technologies in all spheres of science as well as a huge amount of information available, the process of linguistic analysis of the text, and in particular the analysis of the text for semantic unambiguity requires maximal automation.

The research developed a specialized expert system using the C# programming language for providing an automatic survey of native speakers and analyzing lexical meaning of phrases to be implemented as a desktop program for Windows. This takes into account the possibility of considering the determination of the correspondence of several phrases from dictionary files that require diverse expert assessments.

REFERENCES

1. B. Strickland, "Language Reflects 'Core' Cognition: A New Theory About the Origin of Cross-Linguistic Regularities," *Cognitive Science*, vol. 41, pp. 70–101, 2017. Available: <https://onlinelibrary.wiley.com/doi/10.1111/cogs.12332>
2. V. V. Zhukovska, "Cognitive-quantitative parametrization of positional properties of English detached impersonal/non-verbal constructions with explicit subject," *Zakarpatski Filolohichni Studii*, vol. 17, no. 2, pp. 121–128, 2021. [in Ukrainian] Available: <https://dspace.uzhnu.edu.ua/items/54938f63-cea9-43bd-b849-4c925e87811f>
3. V. M. Zayats and M. M. Zayats, "Methods of comparing statistical characteristics in the formation of samples in linguistics," *Scientific Bulletin Journal of Lviv Polytechnic National University "Information Systems and Networks"*, no. 673, pp. 296–305, 2010. Available: <https://ena.lpnu.ua/handle/ntb/6753>
4. Ye. V. Kupriianov, N. S. Uholnikova, and O. M. Yurchenko, *Structural linguistics in theory and practice*. Kharkiv, Ukraine: NTU "KhPI", 2024. [in Ukrainian]. Available: <https://repository.kpi.kharkov.ua/handle/KhPI-Press/76037>
5. W. Zhang, T. Yoshida, and X. Tang, "Text classification based on multi-word with support vector machine," *Knowledge-Based Systems*, vol. 21, no. 8, pp. 879–886, 2008. Available: <https://www.sciencedirect.com/science/article/abs/pii/S0950705108000968?via%3Dihub>
6. A. Sitar, "Statistical analysis of phraseologized sentences: an indicator of mutual information association," *Ukrainian Linguistics*. Kyiv: Taras Shevchenko National University. vol. 1, no. 46, pp. 114–125, 2016. [in Ukrainian] Available: [https://doi.org/10.17721/um/46\(2016\).103-125](https://doi.org/10.17721/um/46(2016).103-125)
7. W. Zhang, T. Yoshida, and X. Tang, "Text classification based on multi-word with support vector machine," *Knowledge-Based Systems*, vol. 21, no. 8, pp. 879–886, 2008. Available: https://www.researchgate.net/publication/222219356_Text_classification_based_on_multi-word_with_support_vector_machine#fullTextFileContent
8. V. Lytvyn, V. Vysotska, and M. H. Hrendus, "Method of data expression from the Ukrainian content based on the ontological approach," *Radio Electronics, Computer Science, Control*, no. 3, vol. 2362, pp. 53–70, 2018. Available: <https://ric.zp.edu.ua/article/view/149991>
9. J. Skeet, *C# in Depth*, 4th ed. Shelter Island, NY: Manning, 2019. Available: <chrome-extension://efaidnbnmnibpcjpcglclefindmkaj/https://dl.ebooksworld.ir/motoman/Manning.Csharp.in.Depth.4th.Edition.www.EBooksWorld.ir.pdf>

Шабельник Тетяна Володимирівна	<i>Доктор економічних наук, професор; завідувачка кафедри економічної кібернетики і системного аналізу, Харківський національний економічний університет імені Семена Кузнеця, просп. Науки, 9-А, Харків, 61166, Україна;</i>
Прокопович Світлана Валеріївна	<i>Кандидат економічних наук, доцент; Доцент кафедри економічної кібернетики і системного аналізу, Харківський національний економічний університет імені Семена Кузнеця, просп. Науки, 9-А, Харків, 61166, Україна;</i>

Експертна система оцінювання закономірностей мови методами непараметричної статистики

Анотація. В роботі аргументовано, що при проведенні лінгвістичних досліджень мають місце завдання, вирішення яких потребує використання інструментарію непараметричних статистичних методів, зокрема проблеми семантичної відмінності речень.

Будь-яка мова, як система комунікації, характеризується поєднанням дискретних одиниць, особливою структурою та елементами, що пов'язаними між собою. Постійне збільшення їхньої кількості є результатом послідовної еволюції та розвитку мови. Використання статистичних методів у лінгвістиці, безумовно, підвищить ефективність лінгвістичного спостереження, його точність та надійність, а також узгодженість та доказовість лінгвістичних припущень.

Мету наукового дослідження визначено як, розробка експертної системи оцінки закономірностей мови методами непараметричної статистики для будь-якої мови.

Методи дослідження. В роботі використано методи непараметричної статистики, нотація IDEF4, мова програмування C#.

В роботі розроблена спеціалізована експертна система з використанням мови програмування C # для здійснення автоматичного опитування «носіїв» мови та аналізу лексичного значення фраз, яка реалізована у вигляді Desktop-програми для ОС Windows.

Для вибору мови програмування експертної системи сформульовані наступні вимоги: програмний продукт повинен бути реалізований для ОС Windows, забезпечувати якісний і доброзичливий інтерфейс, крім того немає необхідності в розподіленому додатку. Програмний продукт не надає можливість редагувати вихідний документ. У програму зчитується текст в форматах docx, rtf або txt. В алгоритмі передбачені можливості створення нового словника фраз із введенням їх експертних оцінок, перегляд і завантаження існуючих словників.

При роботі програми врахована можливість урахування визначення відповідності декількох фраз з файлів словників з різними експертними оцінками.

У лінгвістичних дослідженнях якісних характеристик текстів досить точні результати дають непараметричні критерії відмінностей мовних елементів. В умовах суцільної діджиталізації інформації та її величезних обсягів процес лінгвістичного аналізу тексту, зокрема аналізу тексту на смислову однозначність потребує максимальної автоматизації.

Ключові слова: експертна система, методи непараметричної статистики, закономірності мови, програмне забезпечення.