Міністерство освіти і науки України
Чорноморський національний університет імені Петра Могили
ДНУ «Інститут модернізації змісту освіти»
Південний науковий центр НАН та МОН
Інститут української археографії та джерелознавства
імені М. С. Грушевського НАН України
Первинна профспілкова організація ЧНУ ім. Петра Могили

**«МОГИЛЯНСЬКІ ЧИТАННЯ – 2026:
досвід та тенденції розвитку суспільства в Україні: глобальний, національний та регіональний аспекти»**

XXIX Всеукраїнська науково-практична конференція

ТЕЗИ ДОПОВІДЕЙ

ТЕХНІЧНІ НАУКИ

Миколаїв, 9–13 листопада 2026 року

Миколаїв – 2026

УДК 378(06)ЧНУ
М 74

М 74 Могилянські читання – 2026 : досвід та тенденції розвитку суспільства в Україні : глобальний, національний та регіональний аспекти. Технічні науки : XXIX Всеукр. щоріч. наук.-практ. конф. 9–13 листоп. 2026 р., м. Миколаїв : тези / М-во освіти і науки України ; ЧНУ ім. Петра Могили. Миколаїв : Вид-во ЧНУ ім. Петра Могили, 2026. 376 с.

© ЧНУ ім. Петра Могили, 2026

ЗМІСТ

СЕКЦІЯ: ТЕХНІЧНІ НАУКИ

ПІДСЕКЦІЯ: АВТОМАТИЗАЦІЯ, КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ ТА РОБОТОТЕХНІКА.....12

Денис Гончаров АВТОМАТИЗОВАНЕ РЕЗЕРВНЕ КОПІЮВАННЯ ЖУРНАЛІВ ОЦІНОК MOODLE 5.....12

Юрій Димитров, Олександр Трунов ІНФОРМАЦІЙНА СИСТЕМА ПІДТРИМКИ ДІАГНОСТУВАННЯ НЕСПРАВНОСТЕЙ АВТОМАТИЗОВАНОЇ СКЛАДСЬКОЇ СИСТЕМИ ЗАСОБАМИ ВЕБ-ТЕХНОЛОГІЙ.....17

Владислав Кайда, Денис Гончаров АВТОМАТИЗОВАНА СИСТЕМА КЕРУВАННЯ ПРАЛЬНЕЮ САМООБСЛУГОВУВАННЯ З ОПЛАТОЮ ЧЕРЕЗ МОБІЛЬНИЙ ЗАСТОСУНОК.....20

Леонід Клименко, Микола Сідєлєв, Дмитро Колода, Нікіта Кім ЗАБЕЗПЕЧЕННЯ БЕЗПЕРЕБІЙНОГО ЕЛЕКТРОПОСТАЧАННЯ ОБ'ЄКТІВ ЦИВІЛЬНОЇ ІНФРАСТРУКТУРИ 26

Антон Савчук, Олександр Трунов ПОРІВНЯННЯ ПРИНЦИПІВ РОБОТИ ЗГОРТКОВИХ І ТРАНСФОРМЕРНИХ НЕЙРОННИХ МЕРЕЖ У ЗАДАЧАХ СЕГМЕНТАЦІЇ ІНТЕРФЕЙСІВ АСК РОБОТОТЕХНІЧНИХ СИСТЕМ.....38

Віктор Шевченко, Олег Щесюк РОЗРОБКА СИСТЕМИ АВТОМАТИЧНОГО КЕРУВАННЯ СКЛАДНИМ ТЕХНІЧНИМ ОБ'ЄКТОМ НА БАЗІ НЕЙРОРЕГУЛЯТОРА З ПРОГНОЗОМ 43

Олег Щесюк, Аркадій Драмарецький, Олег Прищепов, Вячеслав Андрєєв, Володимир Шенкевич ІМІТАЦІЙНА МОДЕЛЬ ГІБРИДНОЇ ЕНЕРГЕТИЧНОЇ СИСТЕМИ.....47

ПІДСЕКЦІЯ: КОМП'ЮТЕРНА ІНЖЕНЕРІЯ 52

Валерій Атланов ПРОБЛЕМИ ГЕНЕРАЦІЇ ВОКСЕЛЬНИХ ОБ'ЄКТІВ ТА ШЛЯХИ ЇХ ПОДОЛАННЯ В ЗАВДАННЯХ БІМ-ТЕХНОЛОГІЙ.....52

Юрій Афонін, Андрій Яковішин, Володимир Савінов ЕКСПЕРИМЕНТАЛЬНА ОЦІНКА ЗАСТОСУВАННЯ ГЕНЕРАТИВНОГО ШІ ПРИ ФОРМУВАННІ ГІБРИДНИХ ДАТАСЕТІВ У ЗАДАЧАХ ВИЯВЛЕННЯ ВИБУХОНЕБЕЗПЕЧНИХ ПРЕДМЕТІВ 57

Єлизавета Бугакова, Катерина Обухова АПАРАТНО-ПРОГРАМНИЙ КОМПЛЕКС ДЛЯ ВИЗНАЧЕННЯ СТИГЛОСТІ ПЛОДІВ НА БАЗІ ESP32 64

Владислав Воляник, Сергій Пузирьов СИСТЕМА ДОПОВНЕНОЇ РЕАЛЬНОСТІ ДОРОЖНЬОГО РУХУ НА БАЗІ RASPBERRY PI ТА OPENCV 69

Станіслав Гануліч, Сергій Пузирьов МЕТОДИ ТА ЗАСОБИ КОМП'ЮТЕРНОГО ЗОРУ НА БАЗІ RASPBERRY PI 73

Ілля Гуляєв, Євген Дарнапук АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ EDGE-AI КЕРУВАННЯ МОБІЛЬНОЮ ПЛАТФОРМОЮ НА БАЗІ NVIDIA JETSON NANO 78

Нікіта Гюльмамедов, Іван Бурлаченко СЕГМЕНТАЦІЯ КОЛЬОРІВ НА БАЗІ КОМПОНЕНТІВ МУЛЬТИАГЕНТНИХ КІБЕРФІЗИЧНИХ СИСТЕМ 83

Юрій Івченко, Борис Салтовський ПРОГРАМНО-АПАРАТНИЙ КОМПЛЕКС ДЛЯ ЛОКАЛЬНОГО МЕТЕОРОЛОГІЧНОГО МОНІТОРИНГУ 87

Ярослав Крайник, Андрій Кучинський NARROWBAND AUDIO SIGNAL DETECTION USING UNIFORM CIRCULAR ARRAY 91

Ігор Москаленко, Олексій Тогоєв ПРОГРАМНО-АПАРАТНІ ЗАСОБИ АВТЕНТИФІКАЦІЇ В EDGE-ІНФРАСТРУКТУРІ СИСТЕМ БЕЗПЕКИ НА ОСНОВІ МІКРОКОМП'ЮТЕРА RASPBERRY PI.....95

Данило Невідомий, Сергій Пузирьов РОЗПОДІЛЕНА ХМАРНА СИСТЕМА ПАНОРАМНОГО ВІДЕОСПОСТЕРЕЖЕННЯ ЗА АРХІТЕКТУРОЮ EDGE-TO-CLOUD.....101

Віталій Нікольський, Олександр Юрченко ВИКОРИСТАННЯ ТЕЛЕМЕТРІЇ ДЛЯ ОЦІНЮВАННЯ ЕФЕКТИВНОСТІ СИСТЕМ МОНІТОРИНГУ МЕРЕЖЕВОЇ ІНФРАСТРУКТУРИ 104

Андрій Павлюк, Євген Дарнапук ПРОГРАМНА АРХІТЕКТУРА ТА АЛГОРИТМ ОЦІНЮВАННЯ EDGE-AI КЕРУВАННЯ МОБІЛЬНОЮ ПЛАТФОРМОЮ 109

Поліна Пахомова, Євген Дарнапук ПЛАТФОРМА АВТОМАТИЗОВАНОГО ФОРМУВАННЯ СПЕЦИФІКАЦІЙ ГІБРИДНИХ МЕРЕЖЕВИХ ПРОТОКОЛІВ ІЗ ПОСТКВАНТОВОЮ КРИПТОГРАФІЄЮ.....113

Яніс Смолянік, Сергій Пузирьов АВТОМАТИЗОВАНЕ ВИЗНАЧЕННЯ ВІДСТАНІ ДО ПЕРЕШКОД ЗА МОНОКУЛЯРНИМ ЗОБРАЖЕННЯМ У СИСТЕМАХ КЕРУВАННЯ РУХОМИМИ ОБ'ЄКТАМИ НА БАЗІ RASPBERRY PI.....116

В'ячеслав Старченко ЕТОЛОГІЧНИЙ ПІДХІД ДО РОЗРОБКИ МОДЕЛЕЙ ПОВЕДІНКИ РОБОТІВ.....121

Микола Ткаченко ОБҐРУНТУВАННЯ КРИТЕРІЇВ ВИБОРУ ТА ЕКСПЛУАТАЦІЇ ЕЛЕМЕНТІВ ЖИВЛЕННЯ ДЛЯ БЕЗПІЛОТНИХ ЛІТАЛЬНИХ АПАРАТІВ 127

Єгор Ухань МЕТОДИ ТА ЗАСОБИ ЗАБЕЗПЕЧЕННЯ ЗАХИЩЕНОГО МЕРЕЖЕВОГО З'ЄДНАННЯ НА БАЗІ ТЕХНОЛОГІЙ WIREGUARD І OPENVPN.....133

Євгеній Худолій, Євген Дарнапук РОЛІ ТА ПОВНОВАЖЕННЯ ВИДАВЦІВ ПЕРЕВІРЮВАНИХ ОБЛІКОВИХ ДАНИХ У МІЖДОМЕННІЙ АВТОРИЗАЦІЇ.....140

Дмитро Чередниченко, Євген Дарнапук ОГЛЯД ТЕХНОЛОГІЙ КОМП'ЮТЕРНОГО ЗОРУ В EDGE-AI СИСТЕМАХ.....146

Гліб Чернигін, Ірина Журавська СИСТЕМА АВТОНОМНОГО ПЛАНУВАННЯ І ДИНАМІЧНОГО КОРИГУВАННЯ ЗАВДАНЬ БПЛА НА БАЗІ АГЕНТНОГО ШТУЧНОГО ІНТЕЛЕКТУ.....152

Геннадій Чуйко, Євген Дарнапук, Поліна Кравченко DIAGNOSTICS AND CERTIFICATION OF PAULI QUANTUM CHANNELS: ENGINEERING ASPECTS.....157

Віктор Шевченко, Євген Дарнапук ПРАКТИЧНА ОЦІНКА МАСШТАБОВАНОСТІ ТА СТІЙКОСТІ СИСТЕМИ РОЗПОДІЛЕНОГО ОБМІНУ DICOM-ЗОБРАЖЕННЯМИ.....162

Костянтин Шеремет, Сергій Пузирьов ІНТЕРАКТИВНА МАПА СТАНУ MESH-МЕРЕЖІ РУХОМИХ АВТОНОМНИХ МОДУЛІВ: КОНЦЕПЦІЯ НА ОСНОВІ ESP32 ТА LORA MESH.....167

ПІДСЕКЦІЯ: МОДЕЛІ, МЕТОДИ ТА ЗАСОБИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ.....173

Катерина Антіпова ВИЯВЛЕННЯ ЗГЕНЕРОВАНОГО ТЕКСТУ В УМОВАХ ЗМАГАЛЬНИХ АТАК.....173

Єгор Білий, Світлана Боровльова РОЗРОБКА НАВЧАЛЬНОЇ ІТ-ПЛАТФОРМИ З ВИКОРИСТАННЯМ LLM.....180

Владислав Блідар, Євгеній Стоєв АРХІТЕКТУРА КІМНАТНОГО (ROOM-BASED) МАРШРУТИЗАТОРА ПОВІДОМЛЕНЬ ДЛЯ ВЕБСОКЕТ-СЕРВЕРІВ: ПРОБЛЕМИ МАСШТАБУВАННЯ ТА ШЛЯХИ ЇХ ВИРІШЕННЯ.....183

Олександр Горелов, Марина Фаленкова СИСТЕМА ПРОГНОЗУВАННЯ РИЗИКУ НА ОСНОВІ АНСАМБЛЕВИХ МОДЕЛЕЙ.....186

Андрій Євсєєв, Євген Давиденко ВИКОРИСТАННЯ МАШИННОГО НАВЧАННЯ ТА МНЕМОНІЧНОГО МЕТОДУ ХЕЙЗІГА ДЛЯ РОЗРОБКИ МОБІЛЬНОГО ЗАСТОСУНКУ ВИВЧЕННЯ КАНДЗІ...191

Ігор Кандиба ВИЗНАЧЕННЯ ТОНАЛЬНОСТІ КОМЕНТАРІВ ДО МЕДІАКОНТЕНТУ ВСТУПНОЇ КАМПАНІЇ ЗАКЛАДУ ВИЩОЇ ОСВІТИ ЗАСОБАМИ PYTHON 195

Катерина Кірей МОДЕЛЮВАННЯ ЦИФРОВОГО ПРОФІЛЮ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ НА ОСНОВІ ІНТЕГРОВАНИХ ОСВІТНІХ ДАНИХ.....199

Євгеній Кітань, Катерина Антіпова ВИКОРИСТАННЯ АГЕНТІВ ШТУЧНОГО ІНТЕЛЕКТУ В РОЗРОБЦІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ: ПРАВИЛА, ПРОМПТИ ТА ЗОВНІШНІ ІНСТРУМЕНТИ.....204

Лада Ковшун, Стефанія Бондаренко ІНТЕГРАЦІЯ ШТУЧНОГО ІНТЕЛЕКТУ В ЧАТ ДОПОМОГИ КЛІЄНТАМ.....208

Микита Колесніков, Гліб Горбань ОЦІНКА СТІЙКОСТІ КЛАСИФІКАТОРІВ ШТУЧНОГО КОНТЕНТУ ДО ЦІЛЕСПРЯМОВАНИХ ПРОГРАМНИХ ЗБУРЕНЬ.....214

Софія Крива, Євген Давиденко ІНТЕГРАЦІЯ ІНТЕРАКТИВНИХ ВЕБСЕРВІСІВ У ВИКЛАДАННЯ ІНОЗЕМНИХ МОВ У ЗАКЛАДАХ ЗАГАЛЬНОЇ СЕРЕДНЬОЇ ОСВІТИ 219

Денис Поляк, Альона Швед АВТОМАТИЗОВАНА ІНФОРМАЦІЙНА СИСТЕМА ОПЕРАТИВНОГО ПЛАНУВАННЯ ТА ЗБИРАННЯ ТОМАТІВ 224

Михайло Прозапас, Микита Ковальчук ПРОЄКТУВАННЯ АРХІТЕКТУРИ МОБІЛЬНОЇ СИСТЕМИ ОБЛІКУ ПРОДУКТІВ І ПІДБОРУ РЕЦЕПТІВ 227

Данило Пройдисвіт, Ігор Кандиба ІНСТРУМЕНТАРІЙ РОЗРОБКИ ВЕБЗАСТОСУНКУ АНАЛІЗУ ФІНАНСІВ 230

Віктор Раленко, Євгеній Стоєв АРХІТЕКТУРНІ ПРИНЦИПИ СТВОРЕННЯ ВЛАСНИХ БІБЛІОТЕК РЕАКТИВНИХ КОМПОНЕНТІВ ВЕБІНТЕРФЕЙСІВ 235

Віктор Раленко, Марина Фаленкова ЗАСТОСУВАННЯ МЕХАНІЗМІВ УВАГИ ДЛЯ ДЕТЕКЦІЇ САРКАЗМУ В ТЕКСТІ 239

Анна Садова, Євген Давиденко ЦИФРОВІЗАЦІЯ МЕНЕДЖМЕНТУ КЛІНІНГОВИХ КОМПАНІЙ НА ОСНОВІ ПРОГНОЗУВАННЯ ТА ЕВРИСТИЧНОГО ПЛАНУВАННЯ 244

Дмитро Свінтицький, Євген Давиденко ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ АВТОМАТИЗАЦІЇ ФОРМУВАННЯ ТА ОБЛІКУ НАВЧАЛЬНО-МЕТОДИЧНИХ МАТЕРІАЛІВ 248

Євгеній Стоєв, Віктор Раленко АРХІТЕКТУРНІ КОМПРОМІСИ З AI: ОЦІНЮВАННЯ МАСШТАБОВАНОСТІ, НАДІЙНОСТІ, СКЛАДНОСТІ ТА ВАРТОСТІ.....251

Анастасія Темна, Євгеній Стоєв РЕАЛІЗАЦІЯ КЛІЄНТСЬКОГО МОДУЛЯ ГЕОЛОКАЦІЙНОГО ПОШУКУ ТА РЕКОМЕНДАЦІЙ ОГОЛОШЕНЬ У СИСТЕМІ ДОПОМОГИ ВЕТЕРАНАМ “PORUCH” 256

Микола Фісун, Надія Гончарова ЗАСТОСУВАННЯ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ ДЛЯ ПЕРЕВІРКИ КОРЕКТНОСТІ ГРАМАТИК У ФОРМІ БЕКУСА – НАУРА 261

Аліна Худолій, Євген Давиденко АНАЛІЗ ТОНАЛЬНОСТІ ВІДГУКІВ СПОЖИВАЧІВ МЕТОДАМИ МАШИННОГО НАВЧАННЯ.....265

Альона Швед, Марина Фаленкова СУЧАСНІ НАПРЯМКИ РОЗВИТКУ МЕТОДІВ АНАЛІЗУ ТОНАЛЬНОСТІ ТЕКСТІВ.....268

ПІДСЕКЦІЯ: ІНТЕЛЕКТУАЛЬНІ ІНФОРМАЦІЙНІ СИСТЕМИ 273

Надія Болюбаш СЕМАНТИЧНА СЕГМЕНТАЦІЯ ДЛЯ ІНТЕЛЕКТУАЛЬНОГО МОНІТОРИНГУ СІЛЬСЬКОГОСПОДАРСЬКИХ УГІДЬ ЗА ДАНИМИ БПЛА 273

Оксана Брагінець ЧИСЕЛЬНЕ РОЗВ'ЯЗАННЯ НЕЛІНІЙНИХ РІВНЯНЬ: ГРАФІЧНИЙ МЕТОД ВІДОКРЕМЛЕННЯ КОРЕНІВ ТА МЕТОД ХОРД У MAPLE 280

Олександр Герас, Олексій Козлов, Едуард Лисенков НЕЙРОМЕРЕЖЕВІ МОДЕЛІ ДЛЯ ПРОГНОЗУВАННЯ ТЕХНОЛОГІЙ ПРИГОТУВАННЯ ТА ВЛАСТИВОСТЕЙ ПОЛІМЕРНИХ НАНОКОМПОЗИТІВ.....288

Олександр Гожий, Ірина Калініна ІНТЕЛЕКТУАЛЬНА СИСТЕМА ЕКОЛОГІЧНОГО МОНІТОРИНГУ З УРАХУВАННЯМ ДИСБАЛАНСУ КЛАСІВ 293

Валерій Димо БАГАТОЕТАПНЕ РОЗПІЗНАВАННЯ ПОШКОДЖЕНИХ БУДІВЕЛЬ НА СУПУТНИКОВИХ ЗНІМКАХ ..298

Ірина Калініна, Олександр Гожий СИСТЕМНИЙ ПІДХІД ДО РОЗВ'ЯЗАННЯ ЗАДАЧ КЛАСИФІКАЦІЇ З УРАХУВАННЯМ ДИСБАЛАНСУ КЛАСІВ 302

Олександр Ковалів, Юрій Кондратенко, Євген Сіденко ОПТИМІЗАЦІЯ НЕЙРОМЕРЕЖЕВИХ МОДЕЛЕЙ ДЛЯ ПРОГНОЗУВАННЯ ПОШИРЕННЯ ІНФЕКЦІЙНИХ ЗАХВОРЮВАНЬ НА ОСНОВІ ВІДБОРУ ІНФОРМАТИВНИХ ОЗНАК.....306

Інесса Кулаковська МАТЕМАТИЧНА МОДЕЛЬ НА ОСНОВІ МАТРОЇДІВ ДЛЯ ПІДВИЩЕННЯ СТІЙКОСТІ МОРСЬКИХ МЕРЕЖ ЗВ'ЯЗКУ ТА МОНІТОРИНГУ.....309

Олександр Мещанінов СИНЕРГІЯ ЄДНОСТІ ВИКЛАДАЧІВ ЗІ СТУДЕНТАМИ УНІВЕРСИТЕТУ ІНФОРМАЦІЙНОГО СВІТУ.....313

В'ячеслав Перепелиця, Олексій Козлов ТЕХНОЛОГІЯ ОЦІНЮВАННЯ ЯКОСТІ ВИННОЇ ПРОДУКЦІЇ НА ОСНОВІ МЕТОДІВ ПРОСТОРОВОЇ КЛАСТЕРИЗАЦІЇ ТА АНСАМБЛЕВОГО НАВЧАННЯ.....319

Ганна Петрина, Віктор Гожий ІНФОРМАЦІЙНА СИСТЕМА БАГАТОКАНАЛЬНОЇ АТРИБУЦІЇ ТА ОПТИМІЗАЦІЇ МАРКЕТИНГОВИХ БЮДЖЕТІВ НА ОСНОВІ ГРАФОВИХ МОДЕЛЕЙ.....324

Олег Піонткевич, Олексій Козлов АЛГОРИТМІЧНІ ЗАСАДИ ФОРМУВАННЯ ГІБРИДНОГО НАБОРУ ГЕОДАНИХ ОБ'ЄКТІВ ЗБОРУ ТВЕРДИХ ПОБУТОВИХ ВІДХОДІВ.....328

Максим Салютін, Євген Сіденко КАСКАДНЕ СЕМАНТИЧНЕ УТОЧНЕННЯ РЕЗУЛЬТАТІВ YOLO З ВИКОРИСТАННЯМ VISION-LANGUAGE MODEL ДЛЯ РОЗПІЗНАВАННЯ ВІЙСЬКОВОЇ ТЕХНІКИ.....333

Микита Смоленський, Галина Кондратенко, Євген Сіденко ІДЕМПОТЕНТНА ОБРОБКА RFID-ПОДІЙ У МОБІЛЬНІЙ МЕДИЧНІЙ ІНФОРМАЦІЙНІЙ СИСТЕМІ.....339

ПІДСЕКЦІЯ: ГЕОДЕЗІЯ, ЗЕМЛЕУСТРІЙ, КАДАСТР: ТЕНДЕНЦІЇ РОЗВИТКУ В СУЧАСНИХ УМОВАХ.....344

Олена Лазарєва НЕТВОРКІНГ ЯК ІНСТРУМЕНТ ВЕДЕННЯ ЗЕМЕЛЬНОГО БІЗНЕСУ 344

Ярослав Матвєєв СВІТОВИЙ ДОСВІД ЗБАЛАНСОВАНОГО РОЗВИТКУ ТЕРИТОРІЙ ТА МОЖЛИВОСТІ ЙОГО ВИКОРИСТАННЯ В УКРАЇНСЬКІЙ ПРАКТИЦІ 347

Антон Мельник, Андрій Мась, Марина Ячнюк ІНТЕГРАЦІЯ GNSS, ДЗЗ ТА ВІДКРИТИХ ГІС У ПРАКТИЦІ ГЕОДЕЗИЧНОГО ОБЛІКУ ОБ'ЄКТІВ КРИТИЧНОЇ ІНФРАСТРУКТУРИ.....352

Лев Перович ТОЧНІСТЬ ВИЗНАЧЕННЯ ВЕРТИКАЛЬНОГО ПЕРЕМІЩЕННЯ ҐРУНТОВИХ ВОД.....357

Анастасія Руда ОГЛЯД ПІДХОДІВ ДО ЗАХИСТУ ТА ПРОСТОРОВОГО ЗОНУВАННЯ МОРСЬКОГО УЗБЕРЕЖЖЯ.....360

Олена Лазарєва, Олена Сандольська ОСОБЛИВОСТІ СИТУАЦІЙНОГО ПІДХОДУ У ЗЕМЛЕКОРИСТУВАННІ 365

Сергій Чорний ГЕОПРОСТОРОВИЙ АНАЛІЗ ЕРОЗІЙНОЇ НЕБЕЗПЕКИ ЗЕМЕЛЬ ІЗ ВИКОРИСТАННЯМ МОДЕЛІ RUSLE ТА ГЕОІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ.....371

ТЕХНІЧНІ НАУКИ

ПІДСЕКЦІЯ
АВТОМАТИЗАЦІЯ, КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ ТА РОБОТОТЕХНІКА

УДК 004.056.3

Денис Гончаров
PhD, викладач кафедри автоматизації
та комп'ютерно-інтегрованих технологій
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

АВТОМАТИЗОВАНЕ РЕЗЕРВНЕ КОПІЮВАННЯ ЖУРНАЛІВ ОЦІНОК MOODLE 5

Під час війни все частіше відбуваються тривалі відключення електроенергії, особливо в зимовий період. Інколи це може призводити до некоректного завершення роботи серверного обладнання, на якому розгорнуто систему управління навчанням, яка відіграє важливу роль для забезпечення виконання університетом освітньої діяльності. Однією з таких систем є Moodle 5 Чорноморського національного університету імені Петра Могили. Відновлення працездатності сервера після аварійного вимкнення інколи потребує багато часу. У зв'язку з цим виникає потреба у регулярному резервному копіюванні журналів оцінок студентів для того, щоб забезпечити можливість доступу до актуальних результатів навчання навіть у випадку тимчасової недоступності системи Moodle.

Особливо важливо це під час сесії, коли викладачеві необхідно мати можливість оперативно переглядати та використовувати поточні оцінки студентів. Ручне створення резервних копій залежить від людського фактору: викладач може забути виконати копіювання, не мати для цього часу або втратити доступ до системи через аварійне відключення сервера. У результаті це може ускладнити підсумкове оцінювання студентів. Тому доцільним є автоматизоване створення резервних копій журналів оцінок.

Для виконання поставленої мети було написано Python-скрипт для автоматичного експорту журналів оцінок із Moodle 5. Програма

виконує авторизацію в системі від імені викладача, переходить до сторінки експорту оцінок відповідного курсу, вибирає доступні елементи оцінювання, налаштовує параметри експорту та завантажує журнал у CSV-файл [1].

Для взаємодії з вебінтерфейсом Moodle використано бібліотеку Playwright. Вона дозволяє автоматизувати дії, які зазвичай виконує користувач у браузері: відкриття сторінок, натискання елементів інтерфейсу, вибір параметрів форми та завантаження файлів [2]. Особливістю реалізації стало врахування поведінки Moodle 5, де перед початком експорту необхідно активувати блок «Опції формату експорту». Після його відкриття параметри форми встановлюються програмно. Це дозволяє уникнути багаторазового автоматичного прокручування сторінки та робить процес резервного копіювання більш стабільним.

Для роботи програмного засобу на комп'ютері викладача необхідно встановити Python, а також додаткові бібліотеки Playwright і Keyring. Встановлення виконується з командного рядка Windows за допомогою команди: *python -m pip install playwright keyring.* Оскільки Playwright використовує браузерний рушій для автоматизованої взаємодії з вебсторінками, після встановлення бібліотеки необхідно додатково встановити Chromium: *python -m playwright install chromium.*

У конфігураційній частині Python-скрипта задаються такі параметри: адреса Moodle, облікове ім'я користувача та перелік навчальних курсів, для яких необхідно виконувати резервне копіювання журналів. Для кожного курсу вказується його числовий ідентифікатор та назва дисципліни. Наприклад:

```
COURSES = {
   95: "Технічні засоби автоматизації",
   97: "Мікросхемотехніка та мікропроцесори",
}
```

Ідентифікатор курсу можна визначити безпосередньо з адресного рядка браузера після відкриття відповідної дисципліни в Moodle. Адреса сторінки курсу має приблизно такий вигляд: https://moodle.example.edu/course/view.php?id=95. Числове значення після параметра id= є ідентифікатором курсу. У наведеному прикладі ідентифікатором є число 95. Саме це значення необхідно внести до словника COURSES у програмі.

Для збереження даних авторизації використовується захищене сховище облікових даних Windows [3]. Пароль Moodle не зберігається

у відкритому вигляді безпосередньо у вихідному коді програми. Перед першим використанням виконується команда: *python moodle_backup.py --save-password.*

Після її запуску користувач вводить пароль Moodle, який зберігається у захищеному сховищі Windows. Надалі програма може виконувати автоматичну авторизацію без необхідності повторного введення пароля.

Після початкового налаштування доцільно перевірити роботу системи у видимому режимі браузера. Для цього використовується команда: *python moodle_backup.py --force --show.*

Параметр --force примусово запускає створення резервної копії незалежно від часу попереднього копіювання. Параметр --show відкриває Chromium у видимому режимі, завдяки чому можна спостерігати за виконанням усіх дій: авторизацією, відкриттям сторінки курсу, переходом до експорту, налаштуванням параметрів і завантаженням журналу.

Під час автоматизованого експорту програма вибирає необхідні елементи оцінювання, відкриває блок параметрів формату експорту, налаштовує формат даних та запускає завантаження журналу. У звичайному автоматичному режимі браузер працює у фоновому режимі та не потребує участі користувача.

Після завантаження файл журналу перевіряється на коректність. Програма контролює, чи містить отриманий файл таблицю з даними студентів, та перевіряє структуру CSV-файлу. Це дозволяє уникнути ситуації, коли замість журналу оцінок у резервну копію помилково потрапляє HTML-сторінка або порожній файл.

Успішно завантажений журнал зберігається у ZIP-архіві [4]. Разом із CSV-файлом до архіву додається службовий файл із метаданими, який містить інформацію про курс, дату і час створення резервної копії, кількість записів та контрольну SHA-256 суму вихідного файлу (рис. 1). Контрольна сума може використовуватися для перевірки цілісності збережених даних [5].



Рисунок 1. – Створений архів зі збереженими даними курсу

Для автоматизації процесу резервного копіювання використовується стандартний планувальник завдань Windows [6]. Реєстрація необхідних завдань виконується безпосередньо Python-скриптом за допомогою команди: *python moodle_backup.py --install-task.* В Windows Task Scheduler створюються два окремі завдання: MoodleGradeBackupWeekly та MoodleGradeBackupStartupCheck. Перше завдання, MoodleGradeBackupWeekly, відповідає за планове створення резервної копії у визначений день тижня та час. Наприклад, у конфігурації програми можуть бути задані такі параметри:

BACKUP_DAY = "Friday"
BACKUP_TIME = "18:00"

У такому випадку резервне копіювання автоматично запускається щоп'ятниці о 18:00. За необхідності день тижня та час можуть бути змінені користувачем.

Друге завдання, MoodleGradeBackupStartupCheck, запускається під час входу користувача до Windows. Його призначення полягає у перевірці актуальності останньої успішної резервної копії. У програмі задається максимально допустимий інтервал між копіями, наприклад: *MAX_BACKUP_AGE_DAYS = 7.*

Якщо з моменту останнього успішного резервного копіювання минуло менше семи днів, програма завершує роботу без створення нового архіву. Якщо ж останній бекап був виконаний сім або більше днів тому, автоматично запускається новий експорт журналів.

Такий механізм є важливим у випадках, коли комп'ютер викладача був вимкнений у момент планового резервного копіювання. Наприклад, якщо запланований запуск мав відбутися в п'ятницю о 18:00, але через відсутність електроенергії комп'ютер не працював, після наступного входу до Windows виконується перевірка дати останньої копії. Якщо встановлений семиденний інтервал уже перевищено, резервування запускається автоматично.

Інформація про останнє успішне резервне копіювання зберігається у службовому файлі стану. Для кожного курсу фіксується назва створеного архіву, календарний період та точна дата й час останнього успішного виконання. Це дозволяє програмі коректно визначати необхідність створення нової копії навіть після пропуску планового запуску.

Додатково в процесі роботи ведеться файл журналу backup.log (рис. 2). У ньому фіксуються повідомлення про успішне створення резервних копій, а також помилки, що можуть виникати під час

авторизації, відкриття сторінок Moodle, експорту або збереження файлів. Наявність такого журналу спрощує контроль роботи системи та пошук причин можливих збоїв.



Рисунок 2. – Записи про успішність збереження журналів

У результаті для початкового розгортання програмного засобу користувачеві необхідно виконати лише декілька основних дій: встановити Python та необхідні бібліотеки, вказати ідентифікатори курсів, один раз зберегти пароль Moodle, перевірити працездатність скрипта та зареєструвати автоматичні завдання в Windows Task Scheduler. Після цього резервне копіювання може виконуватися без постійної участі викладача.

Таким чином, запропонований підхід дозволяє зменшити залежність процесу резервного копіювання від людського фактора та забезпечити регулярне збереження журналів оцінок Moodle 5. Наявність локальної резервної копії є корисною у випадку тимчасової недоступності серверної інфраструктури та дозволяє викладачеві продовжувати роботу з результатами оцінювання студентів. Додаткова перевірка під час входу до Windows знижує ймовірність тривалої відсутності актуальної копії даних у разі пропущеного планового запуску.

Список використаних джерел

1. Grade export. *MoodleDocs*. URL: https://docs.moodle.org/500/en/Grade_export

2. Downloads. *Playwright Python*. URL: https://playwright.dev/python/docs/downloads

3. Welcome to keyring documentation! *Keyring documentation*. URL: https://keyring.readthedocs.io/en/latest/
4. Zipfile – Work with ZIP archives. *Python documentation*. URL: https://docs.python.org/3/library/zipfile.html
5. Hashlib – Secure hashes and message digests. *Python documentation*. URL: https://docs.python.org/3/library/hashlib.html
6. Schtasks create. *Microsoft Learn*. URL: https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/schtasks-create

УДК 681.518

Олександр Трунов
доктор технічних наук, професор,
професор кафедри автоматизації
та комп'ютерно-інтегрованих технологій
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Юрій Димитров
викладач кафедри автоматизації
та комп'ютерно-інтегрованих технологій
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ІНФОРМАЦІЙНА СИСТЕМА ПІДТРИМКИ ДІАГНОСТУВАННЯ НЕСПРАВНОСТЕЙ АВТОМАТИЗОВАНОЇ СКЛАДСЬКОЇ СИСТЕМИ ЗАСОБАМИ ВЕБ-ТЕХНОЛОГІЙ

При передачі в експлуатацію автоматизованої системи важливим є питання передачі замовнику технічної документації, а також спеціалізованої інформації про роботу системи. До такої інформації також належить перелік можливих несправностей обладнання, помилок в роботі, а також методів діагностики, вирішення і несправностей і помилок. Чим ближче ця інформація доступна до місця виконання технологічного процесу, тим краще. Тому класично так звані панелі людино-машинного інтерфейсу (далі HMI – англ. Human Machine Interface) встановлюються на шафах керування. Створення таких засобів класичним програмним забезпеченням для

розробки систем автоматизації, зокрема панелей людино-машинного інтерфейсу є досить обмеженим. Незважаючи на високу вартість спеціалізованого програмного забезпечення для розробки, також важливу роль відіграє клас і вартість самої панелі. Таким чином, в панелях нижчого рівня функціонал доступних графічних елементів, засобів створення програмних скриптів візуалізації і кількості мережевих під'єднань є обмеженим. Крім того, при наявності HMI панелей різного виробництва на різних ділянках технологічного процесу зростає складність редагування систем інформаційної підтримки.

В даній роботі запропоновано технічне рішення для створення дружньої до персоналу, уніфікованої і простої в редагуванні системи інформаційної підтримки.

В роботі [1] розглядаються архітектура і аспекти створення веб-сервера та можливості підвищення надійності. За основу при створенні системи взята проста базова архітектура веб-додатку. Веб-додаток розміщено в локальній мережі на одному із комп'ютерів, що обслуговує систему. Комп'ютер до промислової мережі під'єднано через вихід мережевої карти, що призначена для виходу у LAN промислову мережу. На цьому комп'ютері створено і запущено локальний веб-сервер. До цього локального веб-серверу під'єднано по локальній мережі через LAN і WLAN необхідні панелі HMI а також комп'ютер оператора.



Рисунок 1. – Знімок екрану HMI інформаційної системи

Для інформаційної системи зібрано всі коди помилок, описано суть помилки, а також інформацію з необхідними діями, що їх необхідно виконати персоналу. Сама інформаційна сторінка створена у форматі HTML5 і розміщена у папці на комп'ютері. HTML5 дозволяє створювати динамічні елементи і, таким чином, робить інформаційну систему сучасною і зрозумілою персоналу. На рис. 1 можна побачити опис інформаційної системи, перелік помилок з відповідним кодом, поле вводу коду помилки. Для простоти пошуку всі помилки йдуть списком; при натисканні на помилку динамічно розгортається додаткова інформація про помилку. У динамічній інформації вказана конкретна причина і дії оператора, що мають бути виконані при появі даної помилки. Завдяки тому, що інформаційна система – це веб-сторінка, в систему можна додавати велику кількість технічної інформації і подробиць. Крім того, персонал для зручності може змінювати розмір тексту.

Для відображення HTML сторінки на панелі HMI Siemens Basic Panel у у TIA PORTAL створено проєкт графічного дисплею, на якому розміщено стандартний графічний елемент Web-Browser для перегляду веб-сторінок. У налаштуваннях веб-браузеру ми прописуємо IP адресу комп'ютеру, на якому розміщено сервер інформаційної системи, а також шлях до HTML сторінки. На рис. 1 підключення здійснено через спеціальний Sm@rt Server – програмний продукт, що дозволяє віддалене підключення до панелей Siemens по мережі Profinet [2].

Висновки:

1. Запропоноване технічне рішення з переведення інформаційної системи підтримки діагностування несправностей автоматизованої складської системи дозволило скоротити час на розробку і підтримку даної системи, шляхом уніфікації системи за рахунок того, що система може відображатись на панелях HMI різних виробників, поколінь, мобільних пристроїв і локальних операторських комп'ютерів [3], у випадку, якщо панелі оснащені веб-переглядачем.

2. Запропоноване рішення підвищує надійність системи за реалізації дублювання серверної частини.

3. Запропоноване рішення є більш гнучким для розробки сучасних інтерфейсів оператора, що, в свою чергу, підвищує впізнаваність та пришвидшує пошук, діагностування і виправлення помилок технологічної ділянки.

Список використаних джерел

1. Пупена О. М., Ельперін І. В. Промислові мережі та інтеграційні технології в АСУ ТП. Київ : НУХТ, 2017. 301 с.
2. SIMATIC HMI. HMI devices. Basic Panels 2nd Generation : Operating Instructions / Siemens AG. 2016. 132 p. URL: https://assets.new.siemens.com/siemens/assets/api/uuid:91f361e9-9ad2-4f60-a883-aa36aa95b348/basic-hmi-key-panel9.pdf (дата звернення: 08.10.2026).
3. HMI. WinCC (TIA Portal). WinCC Unified Engineering V18 : System Manual / Siemens AG. 11/2022. URL: https://support.industry.siemens.com/dl/files/308/109813308/att_1122197/v1/WinCC_VisualizingProcessesUnified_enUS_en-US.pdf (дата звернення: 10.10.2026).

УДК 004.42:681.5

Владислав Кайда
здобувач першого (бакалаврського) рівня вищої освіти
спеціальності «Автоматизація,
комп'ютерно-інтегровані технології та робототехніка»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Денис Гончаров
PhD, викладач кафедри
автоматизації та комп'ютерно-інтегрованих технологій
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

АВТОМАТИЗОВАНА СИСТЕМА КЕРУВАННЯ ПРАЛЬНЕЮ САМООБСЛУГОВУВАННЯ З ОПЛАТОЮ ЧЕРЕЗ МОБІЛЬНИЙ ЗАСТОСУНОК

У даній роботі розглядається задача автоматизації керування пральними машинами самообслуговування в гуртожитках і житлових комплексах. Актуальність зумовлена потребою в безконтактному запуску обладнання та автоматичному обліку циклів без участі обслуговуючого персоналу.

Мета роботи – розробити систему, яка пов’язує оплату з дозволом на один цикл прання та веде журнал роботи машин. Для мешканця гуртожитку це означає можливість перевірити стан обладнання в телефоні, оплатити послугу й отримати повідомлення про завершення прання.

Для порівняння обрано три способи організації оплати: через працівника, через автономний жетонний автомат без підключення до мережі та через запропонований застосунок. У таблиці 1 наведено їх відмінності за функціями, потрібними для пральні в гуртожитку.

Таблиця 1. – Порівняння способів оплати й обліку роботи пральні

Параметр	**Оплата через працівника**	**Жетонний автомат без мережі**	**Запропонована система**
Дозвіл на прання	Надає працівник	Після приймання жетона	Після перевірки оплати сервером
Облік платежів	Записи або касова програма	За жетонами чи лічильником	Журнал замовлень і платежів
Стан машини в телефоні	Потрібне окреме рішення	Потрібне окреме рішення	Передбачено в застосунку
Залежність нового запуску від Інтернету	Відсутня за готівкової оплати	Відсутня	Потрібний зв’язок із сервером

Порівняння стосується наведених варіантів організації роботи. У запропонованій системі віддалений перегляд стану машини потребує справного мережевого з’єднання. Тому поточний цикл контролюється локально, а нові оплати приймаються лише для доступного обладнання.

Об’єктом автоматизації є пральна машина, яка керується реле через мікроконтролер ESP32. Давач струму визначає стан робочого циклу, а логіка станів (очікування оплати, дозвіл на запуск, контроль циклу, завершення, блокування) реалізована у вигляді кінцевого автомата. Мобільний застосунок виконує роль інтерфейсу оператора та передає команди через хмарний сервіс.

ESP32 обрано через наявність Wi-Fi, аналого-цифрового перетворювача та цифрових входів і виходів [1]. Контролер працює біля машини, тоді як облік замовлень і перевірка платежів виконуються на сервері. Розподіл функцій показано на рисунку 1.



Рисунок 1. – Структура системи керування пральнею

Підключення реле передбачається до штатного інтерфейсу зовнішньої оплати сумісної пральної машини. Параметри сигналу визначаються її документацією. Наприклад, у машинах Electrolux Professional WH6-6 описано різні варіанти входів оплати та зворотних сигналів [2]. Таке підключення потребує узгодження рівнів напруги й гальванічного розділення. Живлення машини під час звичайного завершення циклу не вимикається; блокування дверей залишається під керуванням її штатної автоматики.

Користувач сканує QR-код машини, перевіряє її номер і вартість послуги. Перед переходом до оплати сервер перевіряє доступність машини та тимчасово закріплює її за замовленням. У застосунку мешканець бачить власне замовлення, а адміністратор – стан усіх машин і журнал подій. Платіжні реквізити вводяться на стороні платіжного сервісу.

Дозвіл на запуск формується після серверного підтвердження успішної оплати. Для цього можна використати Callback від LiqPay: сервер отримує статус платежу й перевіряє підпис повідомлення за документацією сервісу [4]. Далі він звіряє номер замовлення, суму та валюту. Самого повідомлення про успіх на екрані телефона для запуску недостатньо.

Кожному замовленню відповідає один дозвіл із власним ідентифікатором. Контролер перевіряє його строк дії та зберігає ознаку використання. Повторна доставка тієї самої команди повертає поточний стан замовлення без нового сигналу запуску. Обмін із сервером передбачається через TLS із перевіркою сертифіката та автентифікацією контролера.

Для контролю струму потрібен ізольований вимірювальний вузол із виходом, узгодженим із входом ESP32. За аналогового підключення для класичного ESP32 використовується ADC1, оскільки робота ADC2 має обмеження під час використання Wi-Fi [3]. Межі струму й тривалість перевірки налаштовуються для обраної машини після запису кількох повних циклів.

Пауза в обертанні барабана або очікування набору води можуть супроводжуватися малим струмом. Тому одиничний спад струму не завершує замовлення. За наявності штатного сигналу роботи програми він використовується разом із вимірюванням струму. Якщо даних недостатньо, машина залишається зайнятою до уточнення її стану.

Переходи між станами наведено на рисунку 2. Після оплати контролер дозволяє запуск лише за готовності машини та закритих дверей. Початок програми підтверджується зворотним сигналом або узгодженими ознаками роботи. Стан дверей визначається окремим сумісним входом, оскільки за струмом його встановити неможливо.



Рисунок 2. – Переходи між станами керування

Передбачено обробку аварійних ситуацій: обрив зв’язку, незавершений цикл, відкриті двері. Для цього алгоритм містить сторожовий таймер і перехід у безпечний стан.

Безпечний стан у цьому алгоритмі означає заборону нового запуску та повідомлення адміністратору. Якщо під час прання зникає лише зв’язок із сервером, штатна програма машини продовжується, а контролер накопичує події локально. Після відновлення з’єднання вони передаються на сервер. Реакції на інші ситуації наведено в таблиці 2.

Таблиця 2. – Реакції на позаштатні ситуації

Ситуація	Що перевіряється	Дія системи
Зв’язок зник до запуску	Наявність чинного дозволу та зв’язку	Новий запуск блокується до звіряння замовлення
Двері відкриті перед запуском	Окремий сигнал дверей	Дозвіл не видається до закриття дверей
Оплату прийнято, запуску немає	Час очікування підтвердження	Замовлення передається на перевірку; нова оплата не потрібна
Цикл триває довше допустимого	Стан програми й час роботи	Повідомлення адміністратору та заборона наступного запуску
Живлення відновилося після збою	Збережене замовлення й стан машини	Автоматичний повтор запуску забороняється до звіряння

Сторожовий таймер контролює виконання програми ESP32. У разі зависання передбачається перезапуск контролера; відповідну реакцію потрібно налаштувати в ESP-IDF [5]. Контроль тривалості прання виконується окремим програмним таймером. Його межа задається за документацією машини з урахуванням обраної програми.

Ідентифікатор замовлення та стан його виконання зберігаються в енергонезалежній пам’яті. Для ESP32 придатна бібліотека NVS, яка зберігає пари «параметр – значення» у флеш-пам’яті [6]. Після перезапуску контролер звіряє ці записи з сервером і фактичним станом машини. За невизначеного результату запуску команда не повторюється автоматично.

Для перевірки макета передбачено серії запусків із нормальною оплатою, повторним повідомленням про неї, відключенням Wi-Fi та перезапуском контролера. Окремо перевіряється розпізнавання пауз усередині програми прання. У журналі фіксуються час підтвердження оплати, фактичний початок і кінець циклу, зміни станів та причина блокування. Оплачені замовлення без виконаного циклу розглядає адміністратор для відновлення послуги або повернення коштів.

Очікувані результати: зменшення часу простою обладнання, автоматичний облік циклів і виручки, зниження витрат на обслуговування.

Кількісну оцінку цих результатів можна отримати після випробування макета. Для цього потрібно визначити частку правильно розпізнаних завершень циклу, кількість хибних блокувань і час відновлення зв’язку. Облік виручки ведеться за підтвердженими платежами з урахуванням повернень, а кількість виконаних прань – за журналом циклів.

Список використаних джерел

1. Espressif Systems. ESP32 Series Datasheet. Version 5.3. URL: https://documentation.espressif.com/esp32_datasheet_en.pdf

2. Electrolux Professional. WH6-6. Installation manual. Розділ 9.4 Functions for I/O-cards. URL: https://tools.electroluxprofessional.com /Mirror/Doc/ELS/IN/IN_438917510EN_WH6-6_EN.pdf

3. Espressif Systems. GPIO & RTC GPIO. ESP-IDF Programming Guide. URL: https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/peripherals/gpio.html

4. LiqPay. Callback. Отримання статусу та інформації про платіж. URL: https://www.liqpay.ua/doc/api/callback

5. Espressif Systems. Watchdogs. ESP-IDF Programming Guide. URL: https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-reference/system/wdts.html

6. Espressif Systems. Non-Volatile Storage Library. ESP-IDF Programming Guide. URL: https://docs.espressif.com/projects/esp-idf/en/stable/esp32/api-reference/storage/nvs_flash.html

УДК 65.011.56

Леонід Клименко
заслужений діяч науки і техніки України,
ректор університету,
доктор технічних наук, професор,
професор кафедри автоматизації
та комп'ютерно-інтегрованих технологій,
Микола Сідєлєв
кандидат технічних наук, доцент,
завідувач кафедри автоматизації
та комп'ютерно-інтегрованих технологій
Дмитро Колода
здобувач другого (магістерського) рівня вищої освіти
спеціальності «Автоматизація,
комп'ютерно-інтегровані технології та робототехніка»
Нікіта Кім
здобувач другого (магістерського) рівня вищої освіти
спеціальності «Автоматизація,
комп'ютерно-інтегровані технології та робототехніка»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ЗАБЕЗПЕЧЕННЯ БЕЗПЕРЕБІЙНОГО ЕЛЕКТРОПОСТАЧАННЯ ОБ'ЄКТІВ ЦИВІЛЬНОЇ ІНФРАСТРУКТУРИ

Використання відновлюваних джерел енергії, зокрема сонячної генерації, є одним із напрямів зменшення витрат на електропостачання об'єктів. Особливо тих, що працюють автономно або мають обмежений доступ до централізованих мереж. Проте виробіток сонячних електростанцій (СЕС) залежить від погодних умов і часу доби, тому сонячна генерація сама по собі не здатна гарантувати безперервне живлення споживачів. Для забезпечення надійності до складу системи вводять акумуляторні батареї (АКБ) та резервна дизель-генераторна установка (ДГУ), яка вмикається у разі недостатньої генерації СЕС і зниження рівня заряду АКБ.

На практиці така гібридна система потребує постійного контролю великої кількості параметрів: потужності генерації, рівня заряду АКБ, величини навантаження, стану обладнання та показників якості електроживлення. У разі ручного або частково автоматизованого керування оператору необхідно постійно стежити за показаннями

приладів, аналізувати поточну ситуацію та своєчасно вмикати або вимикати джерела живлення. Такий підхід збільшує залежність системи від людського фактора, призводить до затримок під час перемикання тощо. Водночас кількість доступних комплексних рішень, що забезпечують автоматичне узгодження роботи СЕС, АКБ та ДГУ з візуалізацією стану системи, залишається обмеженою.

Одним із основних напрямів вирішення зазначеної проблеми є застосування автоматизованих систем керування (АСК), які забезпечують збір інформації про поточний стан об'єкта, її оброблення та формування керувальних впливів. До складу сучасних АСК входять вимірювальні пристрої, виконавчі механізми, засоби зв'язку та технічні засоби керування [1].

Особливого значення для промислової автоматизації набули програмовані логічні контролери (ПЛК), які призначені для роботи безпосередньо з технологічним обладнанням. Вони отримують сигнали від датчиків, виконують програмну обробку отриманої інформації та формують сигнали керування для виконавчих пристроїв.

Застосування ПЛК дає змогу реалізовувати алгоритми керування, змінювати їх програмним способом та забезпечувати взаємодію між окремими елементами автоматизованої системи. Проблема побудови сучасної АСК для гібридної сонячно-дизельної системи, таким чином, полягає у необхідності створення рішення, яке здатне працювати автономно, без постійної участі оператора. Воно повинно забезпечувати узгоджену роботу всіх джерел живлення, надійно отримувати інформацію від об’єкта керування, своєчасно формувати керувальні сигнали та наочно відображати стан системи. Тому доцільним є використання промислових ПЛК як центрального елемента системи керування [2].

Вирішення проблеми способом впровадження АСК на основі промислових ПЛК

Вирішення поставленої проблеми може бути реалізоване шляхом впровадження АСК на основі промислового ПЛК. У такій системі контролер виконує функції центрального пристрою оброблення інформації та керування. На його входи надходять сигнали від датчиків і первинних перетворювачів, після чого відповідно до закладеної програми виконується їх оброблення. За результатами оброблення контролер формує керувальні сигнали, які надходять до виконавчих механізмів. Для гібридної системи це означає, що рішення про те, яке джерело живить навантаження, а також про запуск і зупинку дизель-генератора приймається автоматично, на основі поточних параметрів системи.

Перевагою застосування промислових ПЛК є можливість реалізації різних алгоритмів керування без зміни принципової структури апаратної частини системи. Сучасні ПЛК також передбачають можливість підключення додаткових модулів введення-виведення, засобів операторського керування та промислових мереж.

Програмування ПЛК здійснюється відповідно до міжнародного стандарту IEC 61131-3. Стандарт визначає основні підходи до створення програм для програмованих контролерів і передбачає використання, зокрема, мов LAD, FBD, STL та SFC. Це забезпечує уніфікований підхід до розроблення програмного забезпечення систем автоматизації [3].

Для побудови АСК доцільно передбачити окремі рівні взаємодії: польовий рівень, на якому розташовані датчики та виконавчі механізми; рівень керування, представлений промисловим ПЛК; та рівень оператора, представлений засобами візуалізації (SCADA/HMI). Наявність візуалізації дозволяє оператору не керувати процесом вручну, а лише спостерігати за роботою системи й за потреби втручатися.

Отже, впровадження АСК на основі промислового ПЛК дозволяє перейти від безпосереднього ручного керування до програмно реалізованого алгоритму, за яким гібридна сонячно-дизельна система працює автономно. З урахуванням викладеного, для практичної реалізації запропонованого підходу розроблено функціональну схему системи, склад і принцип дії якої розглядаються далі.

Функціональна схема системи

З метою практичної реалізації запропонованого підходу розроблено функціональну схему АСК гібридної сонячно-дизельної системи електропостачання, наведену на рис. 1. Схема об’єднує два основні блоки генерації, сонячний і паливний, а також засоби автоматизації, що забезпечують їх узгоджену роботу.

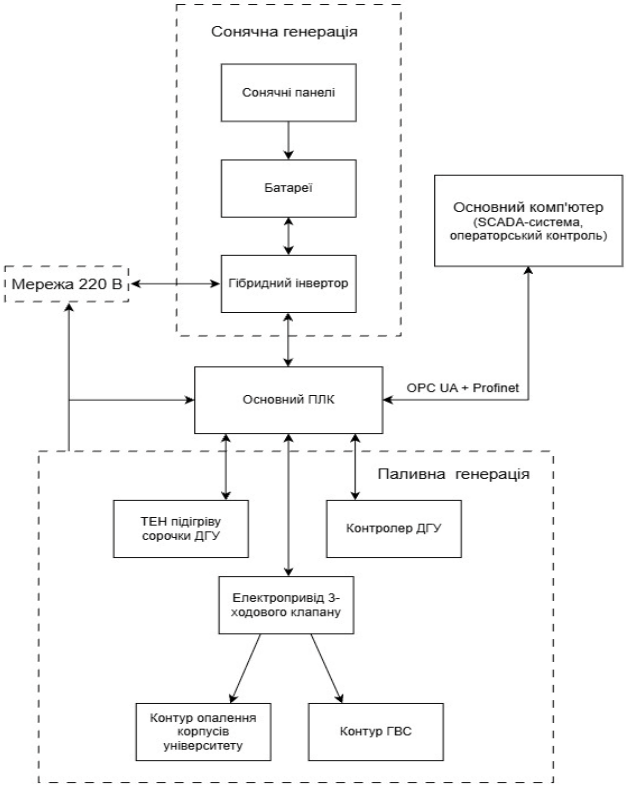


Рисунок 1. – Функціональна схема запропонованої системи

Блок сонячної генерації складається з сонячних панелей, акумуляторних батарей та гібридного інвертора. Сонячні панелі перетворюють енергію випромінювання на електричну й заряджають батареї. Гібридний інвертор забезпечує двосторонній обмін енергією між батареями та мережею 220 В: перетворює постійну напругу батарей на змінну для живлення споживачів і дає змогу виконувати заряд батарей. Інвертор має зв’язок з основним ПЛК, завдяки чому контролер отримує дані про стан цього блоку (наприклад, рівень заряду батарей, потужність генерації й споживання) та може впливати на режим його роботи.

Блок паливної генерації містить дизель-генераторну установку (ДГУ) з власним контролером, ТЕН підігріву сорочки ДГУ,

електропривід триходового клапана та контури теплопостачання. Контролер ДГУ відповідає за безпосереднє керування установкою і передає основному ПЛК інформацію про її стан, а від нього отримує команди запуску та зупинки. ТЕН підігріву сорочки підтримує двигун у прогрітому стані, що забезпечує його готовність до швидкого запуску. Електропривід триходового клапана розподіляє теплову енергію, отриману під час роботи ДГУ, між контуром опалення корпусів університету та контуром гарячого водопостачання (ГВС), що дозволяє додатково використовувати тепло дизельної установки.

Окрім забезпечення електропостачання, у запропонованій системі передбачено утилізацію теплової енергії, що виділяється під час роботи ДГУ. Тепло, яке відводиться від сорочки охолодження двигуна, передається через теплообмінник воді теплового контуру. Додатково вода догрівається за рахунок тепла відпрацьованих газів ДГУ, яке передається їй через другий теплообмінник. Таке двоступеневе використання теплоти двигуна та вихлопних газів дозволяє підвищити загальну енергоефективність установки, оскільки теплова енергія, яка за інших умов була б втрачена, використовується для потреб опалення та гарячого водопостачання. Розподіл нагрітого теплоносія між контуром опалення корпусів університету та контуром ГВС здійснюється за допомогою триходового клапана з електроприводом, яким керує основний ПЛК.

Верхній рівень системи представлений основним комп'ютером зі SCADA-системою. Обмін даними між ним і основним ПЛК виконується за протоколом OPC UA. Це забезпечує візуалізацію стану всіх елементів системи, відображення параметрів у реальному часі, а також можливість операторського контролю.

Узгоджена робота блоків забезпечується основним ПЛК. За нормальних умов основне навантаження беруть на себе сонячні панелі та батареї. У разі зниження рівня заряду батарей до заданого значення за недостатньої генерації СЕС контролер формує команду на запуск ДГУ, яка бере навантаження на себе. Після відновлення умов для роботи сонячної генерації ДГУ зупиняється, а живлення споживачів знову забезпечується від СЕС і батарей. Детальніше структура, можливості та алгоритми керування кожного з блоків розглядаються далі.

Використання дизель-генераторної установки в складі АСК

Паливна генерація (рис. 2) містить дизель-генераторну установку (ДГУ) з власним контролером, ТЕН підігріву сорочки, електропривід триходового клапана, теплообмінники та контури теплопостачання (опалення й ГВС). Основний ПЛК пов’язаний з усіма цими

елементами: обмінюється командами та станом з контролером ДГУ, керує положенням триходового клапана. Крім того, паливний блок з'єднаний з мережею 220 В, через яку ДГУ живить споживачів, коли сонячна генерація й батареї не можуть цього забезпечити.

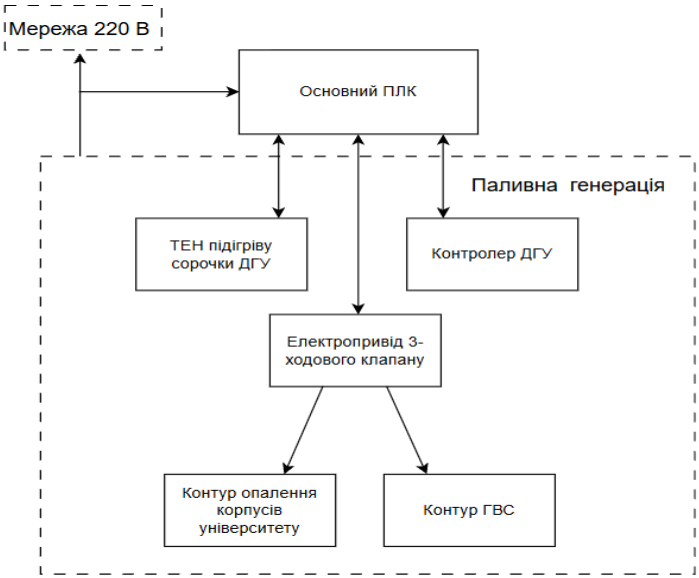


Рисунок 2 – Фрагмент функціональної схеми: паливна генерація

Застосування ДГУ в складі АСК дає змогу автоматизувати запуск, зупинку та контроль установки без участі оператора, підтримувати її в стані готовності до запуску. Також використовувати тепло, що виділяється під час її роботи. До переваг такого рішення належать незалежність від погодних умов і часу доби, стабільне джерело живлення, робота установки лише за потреби (що зменшує витрати палива та моторесурс), швидка готовність завдяки підігріву сорочки та підвищення загальної енергоефективності за рахунок утилізації тепла. Недоліками є витрати палива та необхідність його запасу, шум і викиди. Знос обладнання та потреба в технічному обслуговуванні, а також зниження ефективності роботи двигуна на малих навантаженнях.

Під час роботи ДГУ основний ПЛК бере участь у контролі показників якості електроживлення, насамперед напруги та частоти.

Підключення ДГУ до навантаження виконується лише після виходу на сталий режим. Якщо під час роботи параметри виходять за ці межі, система формує повідомлення оператору або аварійний сигнал. Також ПЛК контролює завантаження ДГУ, не допускаючи її тривалої роботи на навантаженні, що значно відхиляється від оптимального.

Алгоритм керування ДГУ (рис. 3) побудований на аналізі рівня заряду батарей, потужності сонячної генерації та поточного навантаження. У режимі очікування ДГУ вимкнена, а за допомогою ТЕН підтримується задана температура сорочки. Коли рівень заряду батарей знижується до значення запуску, а потужності СЕС недостатньо для живлення споживачів, ПЛК надсилає контролеру ДГУ команду на запуск. Після запуску перевіряється успішність виходу на режим; у разі невдачі виконується обмежена кількість повторних спроб, після чого формується аварійне повідомлення. Після виходу на режим установка бере навантаження на себе, а ПЛК контролює параметри її роботи та керує розподілом тепла між контурами за допомогою триходового клапана. Зупинка ДГУ виконується, коли рівень заряду батарей досягає значення зупинки або сонячна генерація знову здатна забезпечити навантаження. Після зупинки система повертається до режиму очікування [4].

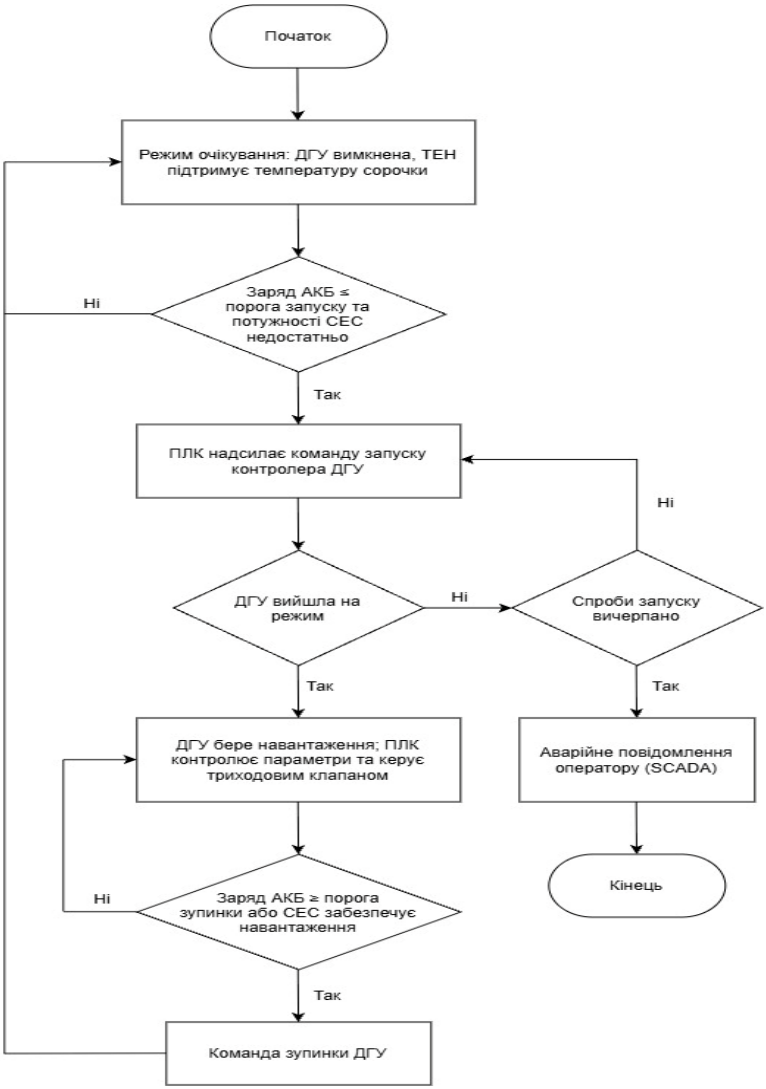


Рисунок 3. – Блок-схема алгоритму керування дизель-генераторною установкою

Використання сонячної генерації в складі АСК

Блок сонячної генерації (рис. 4) складається з сонячних панелей, акумуляторних батарей та гібридного інвертора. Інвертор пов'язаний з основним ПЛК, який отримує від нього дані про стан блока та формує команди керування.

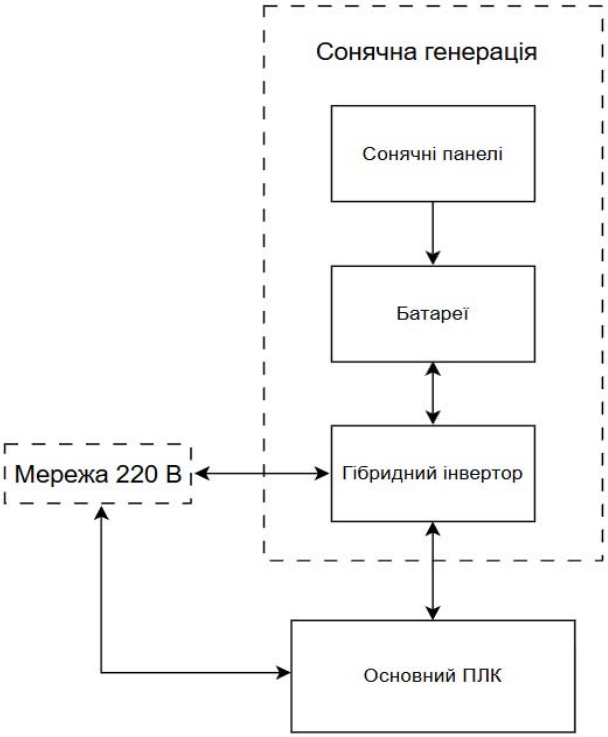


Рисунок 4 .– Фрагмент функціональної схеми: сонячна генерація

Такий склад дозволяє живити навантаження безпосередньо від СЕС, накопичувати надлишок енергії в батареях, забезпечувати споживачів від батарей у відсутність сонячного випромінювання та взаємодіяти з паливною генерацією через спільну мережу. До переваг належать відсутність витрат палива, відсутність шуму й викидів під

час роботи. Недоліками є залежність виробітку від погоди й часу доби, його нерівномірність під час хмарності, обмежені ємність і ресурс батарей, початкові капітальні витрати, а також необхідність контролю заряду й розряду батарей.

Якість електроживлення в цьому блоці значною мірою забезпечується батареями та гібридним інвертором. Батареї компенсують швидкі коливання потужності СЕС, а інвертор підтримує напругу й частоту на виході та регулює обмін енергією з мережею. Основний ПЛК контролює ключові параметри блока (напругу, частоту, потужність, рівень заряду батарей) і впливає на режим його роботи не допускаючи виходу рівня заряду за допустимі межі. За потреби обмежує потужність СЕС (якщо батареї заряджені, а навантаження мале) та у разі дефіциту енергії ініціює запуск ДГУ [5].

Алгоритм (рис. 5) виходить із того, що СЕС є пріоритетним джерелом. Якщо потужності СЕС достатньо для навантаження, споживачі живляться від неї, а надлишок заряджає батареї до верхнього порога заряду. Після його досягнення потужність СЕС обмежується. Якщо потужності СЕС недостатньо (хмарність, вечір, ніч), різницю покриває батарея. Коли рівень її заряду знижується до порога запуску, ПЛК формує команду на запуск ДГУ, і навантаження переходить на паливну генерацію. Коли умови для роботи СЕС відновлюються, а заряд батарей досягає порога зупинки, ДГУ вимикається, і живлення знову забезпечується від СЕС і батарей. Нижній поріг заряду батарей є захисним: у разі його досягнення система формує аварійне повідомлення.

Підсумовуючи, автоматизація гібридних сонячно-дизельних систем електропостачання є одним із важливих напрямів підвищення надійності та енергоефективності автономних і резервних джерел живлення. Запропонована АСК на основі промислового ПЛК, котра поєднує сонячні панелі, акумуляторні батареї, гібридний інвертор і дизель-генераторну установку з SCADA-системою верхнього рівня, дозволяє автоматично узгоджувати роботу джерел. Оцінка запропонованого рішення виявляє помітні переваги у зменшенні втручання оператора, скороченні часу роботи ДГУ та витрат палива та захисті батарей від глибокого розряду. Також у додатковому використанні тепла дизельної установки для потреб опалення та гарячого водопостачання.

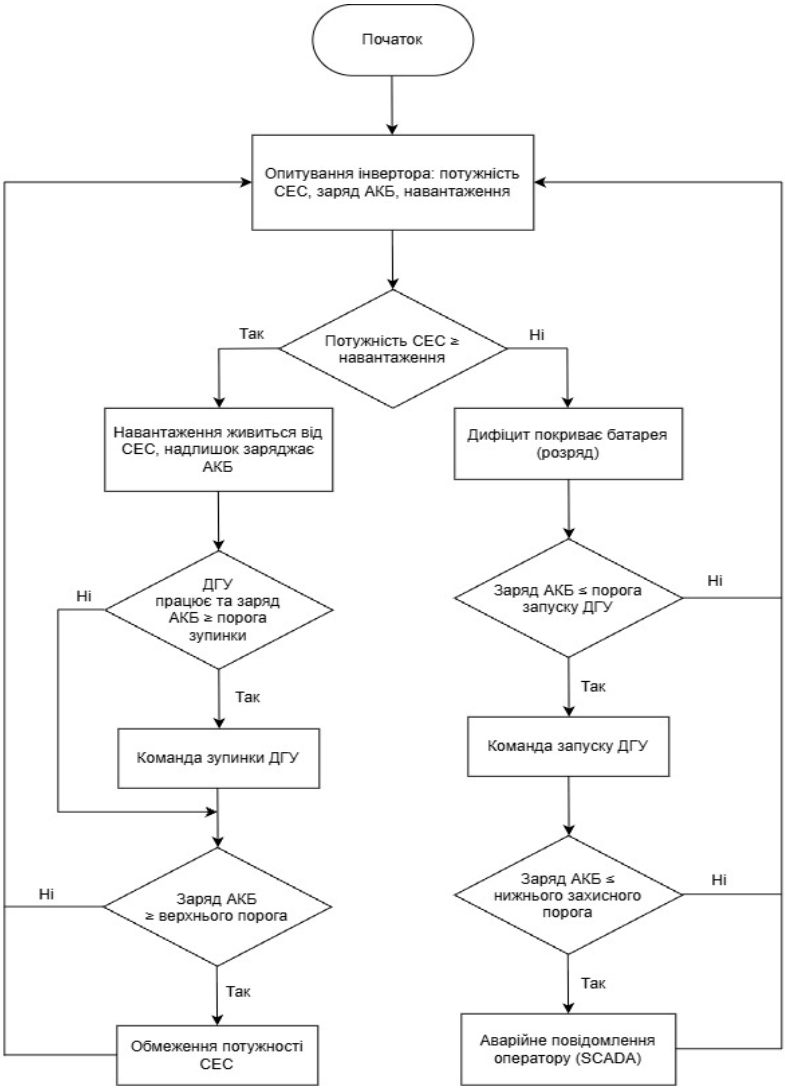


Рисунок 5. – Блок-схема алгоритму керування сонячною генерацією

Однак існують і недоліки, пов’язані з нерівномірністю виробітку СЕС, обмеженими ємністю й ресурсом батарей, витратами палива та необхідністю технічного обслуговування ДГУ. Удосконалення таких

систем – це безперервний процес, націлений на подальшу автоматизацію, підвищення енергоефективності та інтелектуальне керування.

Список використаних джерел

1. Дьомін Д. О. , Пензєв П. С. Автоматизовані системи управління. Харків : Технологічний центр груп, 2024. URL : https://repository.kpi.kharkov.ua/items/dc9e3277-c576-4d72-b09b-9e708da1cfb0

2. Бондаренко І. Н., Бородін А. В., Карнаушенко В. П. Мікропроцесорні системи контролю та керування. ХНУРЕ, 2020. URL : https://openarchive.nure.ua/entities/publication/a3e1c0fd-cd08-461e-8f36-3137e9ca2f72

3. Ritonja J. Automation With Programmable Logic Controllers: Textbook. University of Maribor Press, 2026. URL : https://press.um.si/index.php/ump/en/catalog/book/1133

4. MDPI, Sustainability, т. 11, № 3, ст. 683 («Energy Management and Optimization of a PV/Diesel/Battery Hybrid Energy System Using a Combined Dispatch Strategy»). URL : https://www.mdpi.com/2071-1050/11/3/683

5. MDPI, Energies, 2020, 13(5), 1024 «Addressing Abrupt PV Disturbances, and Mitigating Net Load Profile’s Ramp and Peak Demands, Using Distributed Storage Devices». URL: https://www.mdpi.com/1996-1073/13/5/1024

УДК 004.93

Антон Савчук
аспірант
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Олександр Трунов
доктор технічних наук, професор,
професор кафедри автоматизації
та комп'ютерно-інтегрованих технологій
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ПОРІВНЯННЯ ПРИНЦИПІВ РОБОТИ ЗГОРТКОВИХ І ТРАНСФОРМЕРНИХ НЕЙРОННИХ МЕРЕЖ У ЗАДАЧАХ СЕГМЕНТАЦІЇ ІНТЕРФЕЙСІВ АСК РОБОТОТЕХНІЧНИХ СИСТЕМ

Постановка проблеми. Для автоматизованого оцінювання інтерфейсу автоматизованої системи керування (АСК) робототехнічної системи спочатку потрібно визначити, які елементи зображені на екрані та де вони розташовані. Зображення панелі оператора містить кнопки керування, індикатори стану, числові поля, графіки та мнемосхеми. Помилка у виділенні навіть невеликого елемента може змінити результат подальшої перевірки його помітності, розміру або відстані до сусідніх об'єктів. Тому вибір нейронної мережі для сегментації є окремою дослідницькою задачею, яка передує обчисленню показників якості інтерфейсу.

Дослідження пов'язане з темою дисертації А. Савчука «Підвищення ефективності інтерфейсів АСК робототехнічних систем шляхом автоматизації когнітивних випробувань інноваційних проєктів». У цьому контексті важливо порівняти механізми формування ознак у згорткових і трансформерних мережах та з'ясувати, які властивості цих механізмів слід перевіряти на зображеннях інтерфейсів. Мета роботи полягає в такому порівнянні та розробленні схеми оцінювання придатності моделей до подальших когнітивних випробувань. Метод дослідження – аналітичне зіставлення архітектур U-Net, DeepLabv3+, SegFormer і кодувальника Swin Transformer за першоджерелами [1–5].

Принцип роботи згорткових мереж. Згортковий шар обробляє локальні околи пікселів одним набором навчених ваг. Вихідна активація залежить від значень у цьому околі, зміщення та нелінійної функції. Застосування спільного ядра в різних ділянках зображення дає змогу шукати подібні візуальні ознаки незалежно від їхнього положення. Для шару з одиничним кроком обчислення можна записати формулою (1).

$$h_{i,j,c} = \varphi(b_c + \sum_{m=1}^{M} \sum_{u=-r}^{r} \sum_{v=-r}^{r} w_{u,v,m,c}\, x_{i+u,j+v,m}) \qquad (1)$$

де x – вхідна карта ознак; h – вихідна карта; i, j – просторові координати; c – номер вихідного каналу; m - номер вхідного каналу; M – кількість вхідних каналів; u, v – зміщення всередині ядра; r – його радіус, тобто, розмір ядра становить (2r + 1) × (2r + 1); w – навчені ваги; b – зміщення; φ – функція активації, наприклад ReLU. На наступних шарах операція застосовується вже до карт ознак, сформованих попередніми шарами.

У U-Net просторову деталізацію відновлює декодувальна частина, яка отримує також ознаки від відповідних рівнів кодувальника. Автори визначають її призначення словами «a symmetric expanding path that enables precise localization» [1, с. 234]. Це поєднання деталізованих і узагальнених ознак є підставою розглядати U-Net як кандидата для виділення контурів елементів інтерфейсу. DeepLabv3+ використовує згортки з різними рівнями розрідження та декодер для уточнення меж [2]. Отже, згорткові моделі також можуть враховувати ширший контекст; він залежить від будови всієї мережі. Результати цих праць отримано на інших типах зображень, тому якість на панелях АСК потребує окремої перевірки.

Принцип роботи трансформерних мереж. У трансформері ознаки фрагментів зображення подаються як послідовність векторів токенів [5, розд. 2]. Механізм самоуваги обчислює зв'язки між ними на основі їхнього поточного вмісту. A. Vaswani та співавтори описують вихід уваги як «a weighted sum of the values» [3, розд. 3.2]. Ваги цього підсумовування залежать від сумісності запитів і ключів. Для одного механізму масштабованої скалярної уваги використовується формула (2) [3, розд. 3.2.1].

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \qquad (2)$$

де Q, K, V – матриці запитів, ключів і значень, отримані навченими лінійними перетвореннями векторів токенів; T – транспонування; d_k –

розмірність вектору ключа; QK^T – матриця попарної сумісності запитів і ключів. Функція softmax нормує кожний рядок цієї матриці після масштабування, а множення на V формує вихідні вектори як зважені суми значень. За повної самоуваги кожний токен взаємодіє з усіма токенами; витрати на матрицю сумісності зростають квадратично з їхньою кількістю.

Водночас трансформерні архітектури відрізняються способом організації цих зв'язків. У Swin Transformer увага обмежена локальними вікнами, а їх зсув між блоками забезпечує обмін інформацією між сусідніми вікнами [4]. У задачі сегментації такий кодувальник потрібно поєднати з декодером. SegFormer поєднує ієрархічний трансформерний кодувальник із легким MLP-декодером, який зводить ознаки різних масштабів [5]. Тому протиставлення «локальна згортка – глобальний трансформер» є лише початковим наближенням. Порівнювати слід повні моделі з їхньою конкретною роздільністю ознак і способом відновлення маски.

В обох підходах параметри навчаються за помилкою прогнозування маски. Під час застосування вже навченої моделі ваги згорткового ядра та параметри проєкцій Q, K, V залишаються фіксованими, проте коефіцієнти уваги щоразу обчислюються для поточного зображення. Водночас у трансформерній моделі можуть бути й згорткові операції: наприклад, у блоці Mix-FFN SegFormer використовується згортка 3 × 3 [5]. Отже, порівняння стосується передусім способу взаємодії просторових ознак у моделях.

Порівняння для задач аналізу інтерфейсів. У таблиці 1 зіставлено механізми просторового оброблення ознак та їх значення для сегментації елементів панелі оператора.

Таблиця 1. – Зіставлення механізмів (узагальнення автора за [1–5])

Ознака порівняння	**Згорткові моделі**	**Трансформерні моделі**
Оброблення ознак	Спільні ядра обробляють локальні околи	Ваги уваги залежать від сумісності токенів
Поширення контексту	Через послідовність шарів, масштабування та розріджені згортки	Через повну або обмежену самоувагу, ієрархію та обмін між вікнами
Межі дрібних елементів	Залежать від роздільності ознак і відновлення деталей декодером	Залежать від масштабу токенізації, карт ознак і декодера
Ресурсні витрати	Визначаються ядрами, каналами та роздільністю карт ознак	Залежать від кількості токенів, області уваги та розміру моделі
Що перевіряти на АСК	Збереження тонких меж і розділення сусідніх елементів	Збереження дрібних елементів та стійкість до зміни компонування

Практичний приклад – панель із кількома однаковими кнопками, числовими полями та невеликим аварійним індикатором. Для кнопок потрібно зберегти окремі контури, для індикатора – не втратити малу область під час зменшення зображення. Семантична маска класу «кнопка» сама по собі не гарантує розділення сусідніх екземплярів. Тому для підрахунку елементів слід передбачити перевірку зв'язних компонентів або окремий метод екземплярної сегментації. Прочитання написів і числових значень доцільно виконувати модулем оптичного розпізнавання символів. Ці операції доповнюють сегментацію та дозволяють перевірити, чи відповідає виділена область конкретному елементу керування.

Схема подальшого оцінювання. Пропонуємо зіставляти моделі на єдиному наборі зображень панелей АСК з узгодженими класами та правилами розмітки. Навчальну, валідаційну й тестову частини слід розділяти за проєктами інтерфейсів: майже однакові кадри однієї панелі не повинні потрапляти до різних частин. Крім загальної перевірки, потрібен тест на іншому компонуванні або стилі панелі. Для всіх моделей необхідно зафіксувати роздільність входу, доступні попередньо навчені ваги, бюджет донавчання та апаратну конфігурацію. Так можна відокремити властивості архітектури від переваг додаткових даних або більших обчислювальних ресурсів.

Якість пропонуємо оцінювати за середнім за класами відношенням площі перетину прогнозованої й еталонної масок до площі їх об'єднання (mIoU), повнотою виділення критичних класів і точністю відтворення меж. Для малих індикаторів показники слід подавати окремо: добрий середній результат може приховувати їх пропуск. Також необхідно вимірювати затримку оброблення та споживання пам'яті на цільовому обладнанні. Перед ранжуванням моделей пропонуємо встановити мінімально допустиму повноту виділення аварійних індикаторів і граничну затримку. Значення цих обмежень потрібно обґрунтувати для конкретної АСК. Лише серед моделей, що їх виконують, доцільно порівнювати загальну якість масок та ресурсні витрати.

Отримані маски можна використати для вимірювання площі елементів, їх щільності та взаємного розташування. Проте якісна сегментація ще не є доказом зручності інтерфейсу або зниження когнітивного навантаження оператора. Такі висновки потребують окремих випробувань із завданнями користувача, вимірюванням часу виконання й помилок. У запропонованій схемі сегментація формує відтворювані вхідні дані для цих випробувань.

Висновки. Згорткові моделі послідовно поєднують локальні ознаки, а трансформерні використовують залежні від вмісту зображення ваги уваги. Придатність до сегментації інтерфейсів АСК визначається також масштабом ознак, декодером і умовами навчання. Результатом роботи є зіставлення цих механізмів із вимогами до виділення елементів панелі та схема вибору моделі з урахуванням критичних класів і допустимої затримки. Подальший етап дослідження – експериментальна перевірка схеми на розмічених зображеннях інтерфейсів робототехнічних систем.

Список використаних джерел

1. Ronneberger O., Fischer P., Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation. MICCAI 2015. LNCS. Vol. 9351. P. 234–241. DOI: 10.1007/978-3-319-24574-4_28
2. Chen L.-C., Zhu Y., Papandreou G., Schroff F., Adam H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. Computer Vision – ECCV 2018. LNCS. Vol. 11211. P. 833–851. DOI: 10.1007/978-3-030-01234-2_49
3. Vaswani A., et al. Attention Is All You Need. *Advances in Neural Information Processing Systems.* 2017. Vol. 30. P. 5998–6008. URL: https://papers.neurips.cc/paper/7181-attention-is-all-you-need.pdf
4. Liu Z. et al. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows. ICCV. 2021. P. 10012–10022. URL: https://arxiv.org/abs/2103.14030
5. Xie E., Wang W., Yu Z., Anandkumar A., Alvarez J. M., Luo P. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. *Advances in Neural Information Processing Systems.* 2021. Vol. 34. P. 12077–12090. URL: https://arxiv.org/abs/2105.15203

УДК 681.5

Віктор Шевченко
здобувач другого (магістерського) рівня вищої освіти
спеціальності «Автоматизація,
комп'ютерно-інтегровані технології та робототехніка»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Олег Щесюк
кандидат технічних наук, доцент,
доцент кафедри автоматизації
та комп'ютерно-інтегрованих технологій
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

РОЗРОБКА СИСТЕМИ АВТОМАТИЧНОГО КЕРУВАННЯ СКЛАДНИМ ТЕХНІЧНИМ ОБ'ЄКТОМ НА БАЗІ НЕЙРОРЕГУЛЯТОРА З ПРОГНОЗОМ

Актуальність. Складні технічні об'єкти можуть характеризуватися нелінійністю, інерційністю, запізненням реакції та взаємним впливом технологічних параметрів. За таких умов налаштування регулятора для одного робочого режиму потребують перевірки при зміні навантаження або характеристик об'єкта. Перспективним напрямом є поєднання нейромережевої ідентифікації динаміки [1] з керуванням за прогнозною моделлю (Model Predictive Control, MPC) [2]. Таке поєднання дає змогу формувати керувальний вплив з урахуванням прогнозованої поведінки об'єкта та допустимих меж роботи виконавчих механізмів.

Мета роботи полягає у формуванні структури системи автоматичного керування складним технічним об'єктом на базі нейрорегулятора з прогнозом і визначенні методики її подальшого дослідження. На цьому етапі розглядається теоретико-методична постановка задачі; числові результати моделювання не наводяться. Під нейрорегулятором із прогнозом розуміється поєднання нейромережевої моделі об'єкта та алгоритму оптимізації майбутніх керувальних дій.

Постановка задачі. Для подальшої реалізації пропонується розглянути двозонний тепловий об'єкт із взаємним теплообміном між зонами. Керованими величинами є температури зон, керувальними

впливами – потужності нагрівачів, а збуреннями - зміни температури середовища та теплового навантаження. Такий вибір дозволить дослідити інерційність, міжканальні зв'язки та обмеження потужності. Конкретні параметри об'єкта й період дискретизації необхідно визначити на наступному етапі за фізичною моделлю або експериментальними даними.

Вихідний вектор y(k) містить температури зон, вектор u(k) – потужності нагрівачів, d(k) – вимірювані збурення; k позначає номер такту дискретизації. Необхідно забезпечити відстеження заданої траєкторії r(k), обмеживши амплітуду та швидкість зміни керувальних дій. Невимірювані збурення враховуються через похибку прогнозу та перевірку роботи замкненого контуру в різних сценаріях.

Нейромережева модель. Як базову архітектуру пропонується використати багатошаровий персептрон із часовими затримками входів і виходів. Він реалізує нелінійну авторегресійну модель із зовнішніми входами. Однокроковий прогноз визначається співвідношенням (1):

$$\hat{y}(k+1 \mid k) = f_\theta(y(k), \dots, y(k-n_y+1), u(k), \dots, u(k-n_u+1), d(k)), \quad (1)$$

де f_θ - нейромережа з параметрами θ; n_y, n_u – кількість використаних відліків виходів і входів; $\hat{y}(k+1 \mid k)$ – прогноз виходу на наступний такт за інформацією, доступною в момент k. Для багатокрокового прогнозу модель застосовується рекурсивно: невідомі майбутні виходи замінюються прогнозованими, а майбутні входи задаються оптимізатором. За відсутності прогнозу збурень їх виміряне значення приймається сталим на горизонті; вплив цього припущення перевіряється окремо.

Навчальна вибірка має охоплювати робочий діапазон температур, різні рівні потужності та перехідні режими. Дані пропонується розділяти на навчальні, валідаційні й тестові за окремими часовими інтервалами або експериментами. Нормування сигналів слід виконувати за статистиками навчальної вибірки. Кількість затримок і нейронів доцільно обирати за похибкою багатокрокового прогнозу на валідаційних даних, а тестову вибірку використовувати лише для остаточної оцінки. Це дасть змогу перевірити придатність моделі саме до прогнозування в контурі керування.

Формування керувального впливу. На кожному такті розв'язується задача мінімізації критерію (2), який поєднує точність відстеження та плавність зміни потужності:

$$J = \Sigma_{i=1}^{N} \|\hat{y}(k + i \mid k) - r(k + i)\|_{Q}^{2} + \Sigma_{j=0}^{M-1} \|\Delta u(k + j \mid k)\|_{R}^{2}. \quad (2)$$

Тут N – горизонт прогнозування; M ≤ N – горизонт керування; Q і R – додатно визначені вагові матриці; $\|v\|_Q^2 = v^T Q v$; Δu(k + j | k) – приріст керувального впливу відносно попереднього такту. Перший приріст обчислюється від фактично прикладеного u(k − 1). Після завершення горизонту керування потужність у прогнозі залишається сталою. Ваги визначаються після узгодження масштабів сигналів, щоб різні фізичні одиниці не спотворювали оптимізацію.

Під час оптимізації вводяться обмеження $u_{min} \le u \le u_{max}$ та $\Delta u_{min} \le \Delta u \le \Delta u_{max}$. Для прогнозованих температур задаються технологічно допустимі межі. До об'єкту прикладається лише перший елемент знайденої послідовності керувань; на наступному такті прогноз і розрахунок оновлюються за новими вимірюваннями [3]. Горизонт прогнозування має охоплювати суттєву частину перехідного процесу, а час розв'язання задачі повинен бути меншим за період дискретизації.

Структура системи. Запропонована система включає блок збору й перевірки вимірювань, буфер попередніх відліків, нейромережевий предиктор, оптимізатор, блок контролю допустимості керування та виконавчі механізми. Окремий наглядовий блок перевіряє збіжність оптимізації, час обчислення й відхилення фактичного виходу від прогнозу. У разі втрати допустимого розв'язку або перевищення порога похибки передбачається перехід до резервного регулятора. Пороги перемикання, область застосування резервного режиму та умови повернення до прогнозного керування потребують окремої перевірки.

Початкову версію доцільно реалізувати з навчанням нейромережі поза контуром керування та фіксованими вагами під час випробування. Це спростить відтворення результатів і відокремить вплив якості моделі від впливу її адаптації. Наявність обмежень у прогнозній задачі сама по собі не гарантує їх виконання реальним об'єктом за похибки моделі. Тому необхідні незалежний захист від перегрівання та перевірка запасу до технологічних меж. Стійкість замкненого контуру також не випливає лише з точності навчання нейромережі.

Методика перевірки. Для оцінювання запропонованого рішення планується порівняти його з налаштованим ПІД-регулюванням за однакових початкових умов, завдань і обмежень виконавчих механізмів. Програма випробувань має включати зміну заданих температур, стрибок теплового навантаження, зміну міжзонного теплообміну, шум вимірювань і насичення нагрівачів. Окремо слід

перевірити режими, які не входили до навчальної вибірки, але належать до допустимого діапазону роботи.

Показниками якості пропонується обрати середньоквадратичну похибку відстеження, максимальне перерегулювання, час встановлення, суму абсолютних приростів керування, кількість порушень температурних меж і максимальний час обчислення. Для сценаріїв із шумом потрібні повторні запуски з різними реалізаціями завад. Перевагу нейрорегулятора можна буде встановити лише за результатами цих випробувань з урахуванням точності, обчислювальних витрат і дотримання обмежень.

Висновки. Сформовано проєктну структуру системи керування, у якій нейромережева модель прогнозує динаміку об’єкта, а оптимізатор визначає керувальні дії з урахуванням заданої траєкторії та обмежень. Визначено послідовність підготовки даних, навчання предиктора й перевірки замкненого контуру на прикладі двозонного теплового об’єкта. Практичним результатом цього етапу є постановка задачі для подальшої програмної реалізації та моделювання. Наступним кроком має стати побудова моделі об’єкта і кількісне порівняння регуляторів за запропонованою методикою.

Список використаних джерел

1. Narendra K. S., Parthasarathy K. Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks.* 1990. Vol. 1, No. 1. P. 4–27. DOI: 10.1109/72.80202

2. Rawlings J. B., Mayne D. Q., Diehl M. M. Model Predictive Control: Theory, Computation, and Design. 2nd ed. Nob Hill Publishing, 2017. URL: https://sites.engineering.ucsb.edu/~jbraw/mpc/ (Дата звернення: 19.10.2026).

3. do-mpc. *Basics of model predictive control.* Documentation. URL: https://www.do-mpc.com/en/latest/theory_mpc.html (дата звернення: 19.10.2026).

УДК 621.31

Олег Щесюк
кандидат технічних наук, доцент,
доцент кафедри автоматизації
та комп'ютерно-інтегрованих технологій
Аркадій Драмарецький
здобувач першого (бакалаврського) рівня вищої освіти
спеціальності «Автоматизація,
комп'ютерно-інтегровані технології та робототехніка»
Олег Прищепов
кандидат технічних наук, доцент,
доцент кафедри автоматизації
та комп'ютерно-інтегрованих технологій
Вячеслав Андрєєв
кандидат технічних наук, доцент
Володимир Шенкевич
старший викладач кафедри автоматизації
та комп'ютерно-інтегрованих технологій
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ІМІТАЦІЙНА МОДЕЛЬ ГІБРИДНОЇ ЕНЕРГЕТИЧНОЇ СИСТЕМИ

У роботі [1] представлено автоматизовану систему керування гібридною енергетичною системою на базі сонячної електростанції (СЕС), акумуляторної батареї (АКБ) та дизель-генераторної установки (ДГУ). Розроблена система призначена для забезпечення надійного та безперебійного електропостачання споживачів шляхом автоматичного керування джерелами електричної енергії залежно від поточного режиму роботи та енергетичного балансу системи. Для перевірки працездатності розробленої системи керування було виконано моделювання роботи гібридної енергетичної системи.

Метою моделювання є аналіз взаємодії між СЕС, АКБ та ДГУ в різних режимах експлуатації, а також перевірка правильності функціонування розробленого алгоритму керування [2].

Імітаційна модель дозволяє відтворити процеси генерації, накопичення та споживання електричної енергії без використання реального обладнання. У моделі реалізовано основні елементи гібридної енергетичної системи, а саме сонячну електростанцію,

акумуляторну батарею, дизель-генераторну установку та навантаження. (рис. 1). Взаємодія між елементами здійснюється відповідно до розробленого алгоритму.

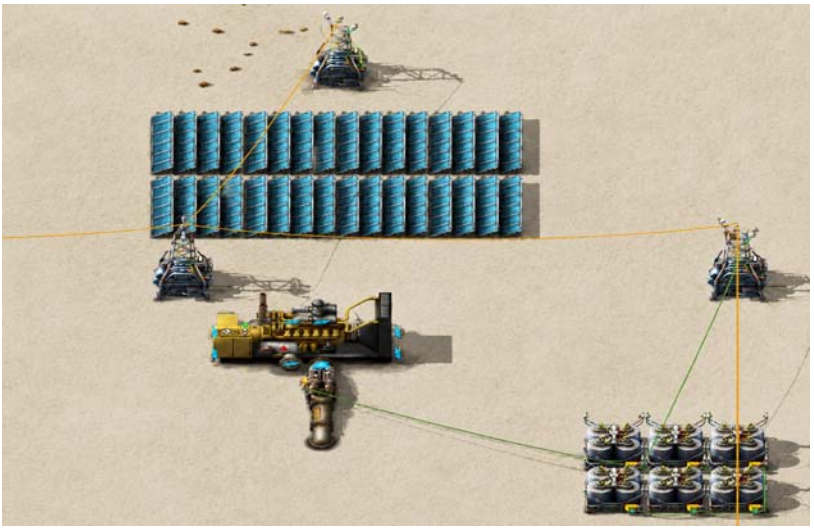


Рисунок 1. – Імітаційна модель гібридної енергетичної системи

Під час моделювання розглядаються різні режими роботи системи. Зокрема, аналізується робота при достатній генерації СЕС, режим живлення від АКБ за недостатньої потужності СЕС та режим резервного електроживлення із залученням ДГУ. Це дозволяє оцінити ефективність роботи системи та перевірити коректність перемикання між джерелами електроенергії.

Особливу увагу приділено перевірці умов автоматичного запуску дизель-генераторної установки при зниженні рівня заряду акумуляторної батареї нижче встановленого порогового значення. Також досліджується процес автоматичного відключення генератора після відновлення необхідного енергетичного балансу системи.

Одним із основних режимів функціонування гібридної енергетичної системи є нормальний режим роботи, за якого СЕС забезпечує достатню генерацію електричної енергії для живлення навантаження та заряджання АКБ (рис. 2.). У цьому режимі ДГУ перебуває у вимкненому стані та не бере участі у забезпеченні енергетичних потреб системи.

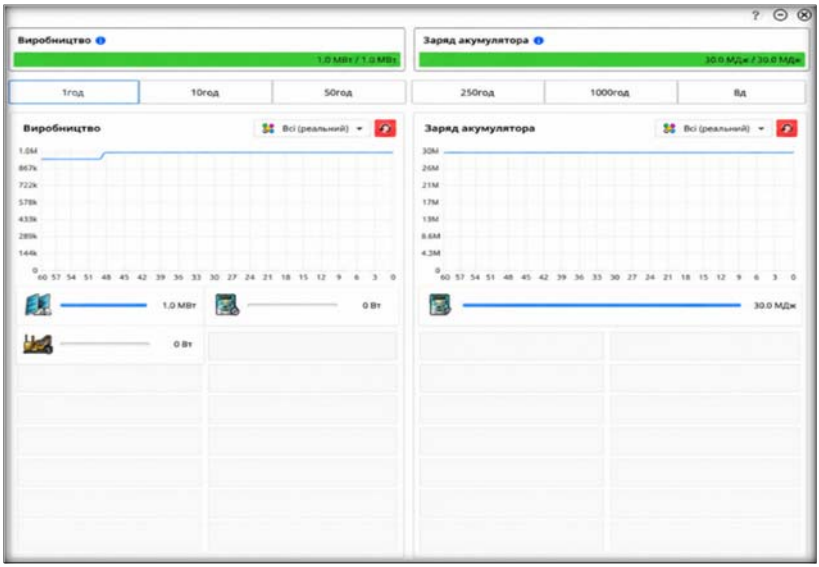


Рисунок 2. – Робота гібридної енергетичної системи
в нормальному режимі

Під час роботи сонячні панелі виробляють електроенергію, яка надходить до споживачів через систему розподілу електричної енергії. У випадку перевищення виробленої потужності над потужністю навантаження надлишок енергії спрямовується на заряд акумуляторної батареї. Такий режим дозволяє максимально використовувати відновлювані джерела енергії та зменшувати витрати на експлуатацію резервного обладнання.

Графічний інтерфейс дозволяє контролювати поточний стан системи, рівень заряду АКБ та вироблену потужність СЕС. При цьому індикатори роботи ДГУ залишаються неактивними, що свідчить про відсутність необхідності використання резервного джерела живлення.

Отримані результати моделювання нормального режиму роботи підтверджують, що за наявності достатньої сонячної генерації система забезпечує стабільне електроживлення навантаження без залучення дизель-генераторної установки, що відповідає розробленому алгоритму керування.

Аварійний режим роботи гібридної енергетичної системи виникає у випадку, коли потужність СЕС є недостатньою для забезпечення

навантаження, а рівень заряду АКБ знижується нижче встановленого порогового значення (рис. 2). У такій ситуації система автоматичного керування формує команду на запуск ДГУ. Після запуску дизель-генератор переходить у робочий режим та бере на себе функцію забезпечення навантаження електричною енергією. Одночасно частина виробленої потужності використовується для відновлення заряду акумуляторної батареї. Це дозволяє підтримувати безперервне електроживлення споживачів навіть за тривалої відсутності достатньої сонячної генерації.

У розробленій моделі аварійний режим імітується шляхом зниження рівня заряду АКБ до критичного значення. Після досягнення встановленого порога система автоматично активує резервне джерело живлення та переводить енергетичний комплекс у режим роботи від ДГУ.

Графічний інтерфейс дозволяє спостерігати процес переходу системи до аварійного режиму. На екрані відображається робота дизель-генератора, зміна потоку електроенергії та поступове збільшення рівня заряду акумуляторної батареї. Завдяки цьому оператор може контролювати процес відновлення енергетичного балансу системи та оцінювати ефективність роботи резервного джерела живлення.



Рисунок 3 – Робота гібридної енергетичної системи в аварійному режимі

Результати моделювання аварійного режиму показали, що система адекватно реагує на зниження рівня заряду АКБ та автоматично запускає резервне джерело живлення. Це забезпечує безперервність електропостачання навантаження та підтверджує працездатність розробленого алгоритму керування.

Результати моделювання режимів роботи дозволяють підтвердити працездатність запропонованої структури гібридної енергетичної системи та ефективність розробленого алгоритму автоматизованого керування. Це створює підґрунтя для подальшого впровадження системи на реальному об'єкті та оцінки її техніко-економічних показників.

Список використаних джерел

1. Щесюк О., Драмарецький А., Андрєєв В.. Автоматизована система керування комплексом дизель-генераторної установки і сонячної електростанції : *Матеріали науково-практичних конференцій Чорноморського національного університету імені Петра Могили. Серія : Дизайн, Комп'ютерні технології : Ольвійський форум – 2026 : стратегії країн Причорноморського регіону в геополітичному просторі :* XXIII міжнар. наук. конф. 29 черв. – 04 лип. 2026 р., м. Миколаїв / М-во освіти і науки України ; ЧНУ імені Петра Могили. Миколаїв : Вид-во ЧНУ ім. Петра Могили, 2026. № 2(1), серія 6. С. 148–152. DOI 10.34132/mspc2026.02.06.33

2. Factorio. Середовище імітаційного моделювання енергетичних систем. URL : https://www.factorio.com (дата звернення: 26.04.2026).

ПІДСЕКЦІЯ
КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

УДК 004.925.8:004.896:721.01

Валерій Атланов
аспірант кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ПРОБЛЕМИ ГЕНЕРАЦІЇ ВОКСЕЛЬНИХ ОБ'ЄКТІВ ТА ШЛЯХИ ЇХ ПОДОЛАННЯ В ЗАВДАННЯХ БІМ-ТЕХНОЛОГІЙ

Вступ та постановка проблеми. Сучасна парадигма цифровізації архітектурно-будівельної галузі (AEC – Architecture, Engineering, and Construction) спирається на концепцію будівельного інформаційного моделювання (Building Information Modeling – BIM). Проте класичне параметричне моделювання в середовищах САПР орієнтоване на ручне компонування заздалегідь визначених типових сімейств або алгоритмічне програмування у візуальних скриптових середовищах. Поява генеративних моделей глибокого навчання (Deep Generative Models) відкриває новий шлях до концептуального пошуку складної форми на основі просторових ескізів, функціональних обмежень або текстових описів. У цьому контексті воксельне представлення 3D-простору виступає найбільш природним і математично прозорим розширенням двовимірної растрової обробки на тривимірний евклідів простір. Воно дозволяє безпосередньо застосовувати перевірений апарат тривимірних згорткових нейронних мереж (3D-CNN), варіаційних автокодувальників (3D-VAE) та генеративно-змагальних мереж (3D-GAN).

Попри фундаментальну математичну узгодженість, пряма інтеграція воксельних структур у практичний робочий процес BIM стикається з глибоким методологічним та технологічним розривом. Якщо для комп'ютерної графіки та ігрових рушіїв достатньо отримати візуально правдоподібний трикутний меш, то в парадигмі BIM геометрична форма є вторинною щодо семантичного змісту. Будівельний елемент повинен мати параметричні зв'язки, топологічну неперервність, чітко визначені планарні межі та строго відповідати міжнародному стандарту відкритого обміну даними IFC (Industry

Foundation Classes, ISO 16739-1). Метою дослідження є комплексна систематизація проблеми синтезу воксельних структур для задач AEC та обґрунтування наскрізної архітектури їх перетворення на параметричні BIM-компоненти.

Аналіз останніх досліджень і публікацій. Питання об'ємної генерації тривимірних форм нейромережами активно розвивається з моменту появи піонерських робіт над архітектурами 3D ShapeNets та 3D-GAN. Дослідники продемонстрували принципову здатність згорткових мереж вивчати латентні простори форм і синтезувати нові об'єкти у вигляді матриць просторової зайнятості. Проте в класичних експериментах генерація зазвичай обмежується ізольованими об'єктами промислового дизайну (меблі, транспортні засоби) у вкрай низькій роздільній здатності – від 32^3 до 128^3 комірок. Масштаб архітектурних комплексів вимагає на порядки вищої просторової роздільної здатності та ієрархічної координації десятків інженерних підсистем.

Останніми роками для подолання бар'єрів низької дискретизації інтенсивно розвиваються неявні нейронні репрезентації (Neural Implicit Representations), зокрема неперервні функції знакових відстаней (Signed Distance Functions – SDF, DeepSDF) та генеративні поля випромінювання. Проте фундаментальні праці у сфері будівельної інформатики (Eastman, et al., Sacks, et al.) доводять: неперервна гладка поверхня, вилучена з нейромережі, залишається геометрично «порожньою» з інженерної точки зору. Без семантичного розчленування на конструктивні компоненти (колони, плити перекриття, зовнішні огороджувальні конструкції) модель не має практичної цінності для розрахункових комплексів та будівельного моніторингу.

Систематизація проблем генерації воксельних архітектурних форм. Комплексний аналіз взаємодії генеративних воксельних моделей та інженерних САПР-систем дозволяє структурувати чотири головні групи науково-технічних проблем:

1. Прокляття кубічної розмірності та обмеження VRAM. Просторова дискретизація тривимірного об'єму регулярною матрицею $N \times N \times N$ демонструє кубічне зростання вимог до обчислювальної пам'яті $O(N^3)$. Якщо стандартна сітка 128^3 потребує близько 2 мільйонів вокселів, то роздільна здатність 512^3, необхідна для мінімального відтворення архітектурних отворів і деталей, вимагає вже понад 134 мільйони комірок. Графічні прискорювачі зазнають фатального переповнення пам'яті (Out-Of-Memory) вже на етапі зворотного поширення помилки при навчання генератора. Вимушене

зниження роздільної здатності викликає грубі артефакти дискретизації («східчастий» аліасинг похилих граней).

2. Топологічна неструктурованість та надлишкова щільність геометрії. Традиційні алгоритми ізоповерхневого вилучення (зокрема класичний алгоритм Marching Cubes) створюють неструктуровані тріангульовані оболонки. Замість єдиної планарної площини стіни формуються тисячі хаотично зорієнтованих трикутників. Такий надлишковий меш не містить інформації про контурні ребра, кути сполучення та координаційні осі, що призводить до паралічу графічного ядра САПР (Revit, ArchiCAD) під час спроби імпорту.

3. Семантичний вакуум (Semantic Gap). Нейронна мережа оперує безіменними числовими значеннями густини речовини або ймовірністю її знаходження в точці простору. У згенерованому масиві відсутній функціональний розподіл: нейромережа «не знає», де розташована несуча колона, де теплоізоляція, а де вентиляційний канал. Відсутність атрибутивних даних робить неможливим призначення матеріалів, розрахунок міцності чи формування специфікацій за стандартами BIM.

4. Порушення будівельної конструктивної логіки. Архітектурна форма підпорядкована законам статики, сили тяжіння та функціонального зонування. Вільна дифузія вокселів у неконтрольованому латентному просторі часто утворює фізично нестійкі консолі, фрагменти, що висять у повітрі без опор, або порушення вертикальної співвісності несучих ядер споруди.

Обґрунтування архітектури запропонованого технологічного конвеєра. Для подолання виявлених протиріч розроблено концептуальний науковий конвеєр трансформації генеративних об’ємних даних у семантичні моделі BIM. Архітектура конвеєра базується на трьох взаємопов’язаних технологічних рівнях:

1. Адаптивні деревоподібні структури (Sparse Voxel Octrees – SVO). Для нейтралізації кубічної складності замість суцільної регулярної сітки використовується розріджене октодерево. Просторовий пошук форми виконується з динамічним дробленням комірок виключно в приповерхневому шарі (Boundary Layer), тоді як однорідний внутрішній масив і вільний повітряний простір кодуються укрупненими вузлами. Це скорочує просторову складність до квазіквадратичної $O(N^2)$, дозволяючи синтезувати моделі еквівалентної деталізації 1024^3 на типовому обладнанні.

2. Гібридне полігональне відновлення (Dual Contouring + RANSAC). Для ліквідації дефектів аліасингу замість Marching Cubes використовується метод Dual Contouring з розрахунком функцій

квадратичної похибки (QEF) на основі нормалей поверхонь. Це забезпечує точне збереження гострих ребер та фасок. Наступний модуль на основі алгоритму RANSAC виконує автоматичне розпізнавання копланарних граней, злиття фасеток у єдині планарні полігони та прив'язку горизонтальних площин до рівнів проектних позначок поверхів.

3. Семантична 3D-сегментація та автоматичне генерування сутностей IFC (Voxel-to-BIM Mapper). Згенерований воксельний масив пропускається через спеціалізовану згорткову мережу семантичної сегментації, яка класифікує вокселі за будівельними класами: фундамент, колона, стіна, перекриття, дверний або віконний отвір. Завершальний алгоритмічний блок генерує параметричну геометрію граничного представлення (B-Rep) та експортує її у відкритий формат IFC4 через класи IfcWallStandardCase, IfcSlab, IfcColumn, IfcBeam із збереженням ієрархії.

Таблиця 1. – Порівняльний аналіз підходів до обробки генеративних 3D-структур у задачах САПР

Критерій аналізу	**Традиційна воксельна модель (Marching Cubes)**	**Запропонований гібридний конвеєр (SVO + Dual Contouring + IFC)**
Просторова складність	Кубічна $O(N^3)$, жорстке обмеження сітки	Адаптивна $O(N^2)$, деталізація межового шару
Топологічна структура	Хаотична тріангуляція, відсутність ребер	Планарні грані, реконструйовані кути та грані
Семантичний статус	Бінарна зайнятість простору [0; 1]	Повна класифікація будівельних елементів
BIM-інтеграція	Відсутня (лише як глухий статичний меш)	Повна параметрична сумісність за стандартом IFC

Оцінка ефективності запропонованих рішень. Проведені розрахунки та первинні експерименти на тестових архітектурних масивах підтверджують високу ефективність запропонованого гібридного підходу. Використання адаптивних октодерев дозволяє знизити пікове навантаження на пам’ять графічного адаптера на 60-75% у порівнянні з монолітною сіткою еквівалентного масштабу. Алгоритмічний модуль планарної децимації забезпечує скорочення надлишкових полігонів на 80-85%, зберігаючи проєктну геометрію та формуючи чисті, придатні для параметричного редагування вузли. Автоматично згенерований IFC-файл без втрат відкриється в

середовищах Autodesk Revit та ArchiCAD як цілісний концептуальний BIM-проєкт рівня деталізації LOD 200.

Висновки та перспективи подальших досліджень. У роботі здійснено комплексний науково-теоретичний аналіз ключових протиріч, що виникають при застосуванні генеративних воксельних моделей штучного інтелекту в інформаційному моделюванні будівель. Доведено, що головними бар'єрами є кубічне зростання ресурсоємності, надлишкова неструктурованість полігональних сіток та семантичний розрив стандартизації. Запропонована комплексна методологія на основі розріджених октодерев, подвійного контурування та семантичної сегментації дозволяє подолати ці бар'єри та перетворити абстрактний воксельний масив на повноцінний BIM-об'єкт. Отримані результати формують основу для створення спеціалізованого хмарного генеративного сервісу, здатного оптимізувати передпроєктні пошуки та відчутно підвищити ефективність і швидкість роботи архітектурних бюро в середовищі BIM.

Список використаних джерел

1. Wu Z., Song S., Khosla A., Yu F., Zhang L., Tang X., Xiao J. 3D ShapeNets: A deep representation for volumetric shapes. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2015. P. 1912–1920.
2. Wu J., Zhang C., Xue T., Freeman B., Tenenbaum J. Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling. *Advances in Neural Information Processing Systems (NeurIPS).* 2016. Vol. 29. P. 82–90.
3. Eastman C., Teicholz P., Sacks R., Liston K. BIM Handbook: A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors. 3rd ed. Hoboken : John Wiley & Sons, 2018. 680 p.
4. Sacks R., Girolami M., Brilakis I. Building Information Modelling, artificial intelligence and construction tech. *Developments in the Built Environment.* 2020. Vol. 4. P. 100011.
5. Ju T., Losasso F., Schaefer S., Warren J. Dual contouring of hermite data. *ACM Transactions on Graphics (TOG).* 2002. Vol. 21, No. 3. P. 339–346.
6. Park J. J., Florence P., Straub J., Newcombe R., Lovegrove S. DeepSDF: Learning continuous signed distance functions for shape representation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* 2019. P. 165–174.

УДК 004.85:004.93

Юрій Афонін
аспірант спеціальності «Комп'ютерна інженерія»
Андрій Яковішин
аспірант спеціальності «Комп'ютерна інженерія»
Володимир Савінов
кандидат технічних наук, доцент,
доцент кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ЕКСПЕРИМЕНТАЛЬНА ОЦІНКА ЗАСТОСУВАННЯ ГЕНЕРАТИВНОГО ШІ ПРИ ФОРМУВАННІ ГІБРИДНИХ ДАТАСЕТІВ У ЗАДАЧАХ ВИЯВЛЕННЯ ВИБУХОНЕБЕЗПЕЧНИХ ПРЕДМЕТІВ

Автоматизація процесів нетехнічного та технічного обстеження територій за допомогою безпілотних літальних апаратів (БпЛА) та наземних роботизованих комплексів (НРК), оснащених системами комп'ютерного зору (англ. Computer Vision, CV), є одним із пріоритетних напрямків досліджень, спрямованих на підвищення безпеки фахівців-демінерів при виконанні завдань гуманітарного розмінування. Застосування НРК у поєднанні зі штучним інтелектом (ШІ) дозволяє локалізувати та візуально розпізнавати небезпечні об'єкти прихованої або замаскованої форми, однак ключовим обмеженням залишається якість і обсяг навчальних візуальних даних [1]. Для навчання нейромережевих моделей детекції об'єктів потрібні масштабні, якісно анотовані набори даних (датасети), які охоплюють різноманітні ракурси, освітлення, ступені заглиблення та тип середовища для різноманітних вибухонебезпечних предметів (ВНП). Для задач автоматизованого візуального розпізнавання активно застосовуються новітні архітектури глибоких згорткових нейронних мереж (англ. Deep Convolutional Neural Network, D-CNN), зокрема найновіша архітектура YOLOv26, яка завдяки блокам C3k2, C2PSA та механізмам позиційно-чутливої уваги забезпечує високий баланс швидкодії та точності [2].

Проте збір реальних зображень ВНП для навчання пов'язаний із безпосереднім ризиком для життя демінерів, обмеженістю доступу до зон бойових дій та частим дефіцитом відкритих навчальних вибірок (англ. data scarcity), особливо для новітніх типів ВНП. У зв'язку з цим

особливо актуальною є розробка і використання новітніх методів отримання тренувальних даних.

З метою подолання дефіциту даних для навчання моделей CV використовують гібридні (комбіновані) датасети, що поєднують синтетичні і реальні зображення. Традиційно, для створення синтетичних зображень використовують ігрові рушії та симулятори. Наприклад, у роботі Дж. МакМіллена та ін. представлено пайплайн SegGen на базі ігрового рушія Unreal Engine 5 для генерації мультимодальних процедурних середовищ, що є основою при створенні синтетичних зображень цільових об'єктів для тренувальних датасетів [3]. Головним недоліком такого способу отримання зображень виступає розбіжність домену (англ. Domain gap) між синтетичними і реальними зображеннями. Одним із підходів до зменшення такої розбіжності, що призводить до покращення валідаційних метрик моделей CV, є модифікація і наближення 3D-сцен до реального домену (реалістичні текстури, геометрія, позиціонування, освітлення тощо). Незважаючи на переваги 3D-рендерингу та симуляцій, формування таких реалістичних 3D-сцен вимагає значних часових і ресурсних затрат на моделювання мешів цільових об'єктів, процедурну генерацію, текстурування об'єктів, накладання матеріалів тощо.

Альтернативним підходом отримання синтетичних (генерованих) зображень сьогодні є використання мультимодальних генеративних та дифузійних (англ. Diffusion models) моделей ШІ, які дозволяють за короткий проміжок часу генерувати або змінювати зображення у різних просторових контекстах за допомогою алгоритмів Text-to-Image та Image-to-Image. Дослідження Н. Мітала та ін. підтверджує, що використання методів пояснювального ШІ (англ. Explainable AI) для модифікації синтетичних даних дозволяє керувати увагою моделей детекції, причому як підвищення, так і контрольоване зниження реалістичності окремих елементів на зображеннях може покращувати узагальнювальну здатність CV-моделей [4].

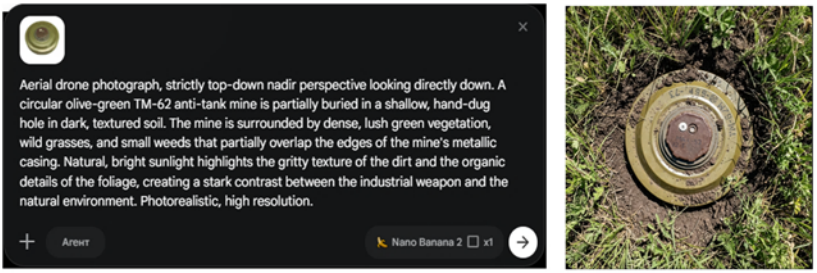
Метою даного дослідження є експериментальне підтвердження гіпотези щодо доцільності використання генеративного ШІ для формування гібридних датасетів та визначення впливу синтетичної складової на валідаційні метрики CV-моделей при виявленні ВНП на тестових вибірках реальних зображень [5]. Як цільовий об'єкт для дослідження обрано протитанкову міну ТМ-62 – один із наймасовіших і найнебезпечніших типів ВНП на території України. Після аналізу найбільш ефективних генеративних моделей ШІ для створення

зображень вибірки використано хмарну мультимодальну модель Gemini 3.1 Flash Image (*Nano Banana 2*) [6].

Генерація передбачає подачу опорного реального зображення міни ТМ-62 без тла і задання текстових інструкцій (англ. text prompt), які детально описують навколишнє середовище (суха трава, зелений газон, опале листя, сніговий покрив, пісок тощо), кути повороту, ракурси та ступінь заглиблення в ґрунт. Таким чином алгоритм включає такі основні кроки:

1. Зразок зображення для генерації;
2. Текстовий промт із заданими ключовими ознаками для генерації;
3. Вибір моделі генерації (*на прикладі Nano Banana 2*) та налаштувань (*к-ть зображень, формат*);
4. Генерація зображень;
5. Валідація людиною якості зображення;
6. Збереження зображення.

На рис. 1 показано результат генерації. Для усіх інших зображень кроки ідентичні.



а) б)

Рисунок 1. – Результат генерації: а) опорне зображення ВНП і текстовий промт; б) результат генерації

$$N_{gen} = 276 \quad (1)$$

$$N_{real_1} = 61 \quad (2)$$

$$N_{real_2} = 56 \quad (3)$$

Всього було згенеровано (1) синтетичних зображень. З відкритих джерел було відібрано (2) реальних зображень міни ТМ-62. Анотування візуальних даних (bbox-розмітка) виконано в середовищі

CVAT. Під час розмітки дотримано контроль якості: зразки із суттєвими артефактами або спотворенням геометрії об'єкта відсіювалися або покращувались за допомогою ШІ.

Для перевірки авторської гіпотези датасет було розділено на 3 варіанти комбінацій вибірок (табл. 1). Важливо, що в усіх трьох варіантах тестова вибірка складалася виключно з реальних зображень, оскільки кінцевою метою є детекція об'єктів у реальному світі, використання синтетичних даних у тестовій вибірці є некоректним з точки зору цінності кінцевих метрик ефективності CV-моделей.

Таблиця 1. – Комбінації вибірок, при N_{gen} (1), N_{real_1} (2), *для варіантів № 2 і № 3 N_{real_2} (3)

№ варіанту	**train-вибірка**			**val-вибірка**			**test-вибірка**	
	K_{train}, од.	частка від N_{real_1}, %	частка від N_{gen}, %	K_{val}, од.	частка від N_{real_2}, %	частка від N_{gen}, %	K_{test}, од.	частка від N_{real_2}, %
№ 1	210	0,0	80,0	66	0,0	20,0	61	100,0
*№ 2	248	67	80,0	75	16,5	20,0	9	16,5
*№ 3	285	16,5	100,0	38	67,0	0,0	9	16,5

Для усіх комбінацій при навчанні використано модель YOLOv26m зі стандартними гіперпараметрами (*за замовчуванням*). У табл. 2 наведено отримані метрики навчання моделей.

Таблиця 2. – Отримані основні метрики навчання CV-моделей

Метрика	**Варіант № 1**	**Варіант № 2**	**Варіант № 3**
Precision	0,89	0,90	0,83
Recall	0,95	0,93	0,71
mAP50	0,97	0,96	0,80
mAP50-95	0,84	0,83	0,65

Detection rate, % (зображень)	80,3 (49/61)	100 (9/9)	100 (9/9)

Аналіз отриманих під час експериментів метрик показав:

1) Варіант № 1 (*Zero-Shot Sim2Real Transfer*) – модель, навчена і валідована виключно на згенерованих зображеннях змогла успішно виявити 80,3 % мін ТМ-62 на реальних тестових зображеннях при точності $Precision = 0{,}89$ та $mAP = 0{,}97$.

2) Варіант № 2 – модель, із поєднанням 80% синтетичних та 67% реальних даних у навчальній вибірці, 20 % і 16,5 % у валідаційній відповідно, забезпечила 100 % детекції на реальних тестових зображеннях із точністю $Precision = 0{,}9$ при $mAP = 0{,}96$.

3) Варіант № 3 – модель, метрики якої продемонстрували достатню точність у $Precision = 0{,}83$, проте валідація суто на реальних даних при 100 % синтетичному тренувальному наборі виявила стрибки в показниках $Recall = 0{,}71$ та обох показниках mAP, що підкреслює наявність значної розбіжності доменів між вибірками та вказує на важливість присутності реальних зразків у валідаційній вибірці.

Отримані метрики показують, що підмішування реальних зображень у тренувальну і валідаційну вибірки дозволяє згладжувати розбіжності між доменами і збалансувати вплив артефактів генерованих зображень.

На рис. 2 показані результати виявлення ВНП для усіх трьох комбінацій на однакових реальних зображеннях із тестових вибірок. Найкращі результати детекції були отримані при перевірці варіанту № 2 комбінації вибірок.

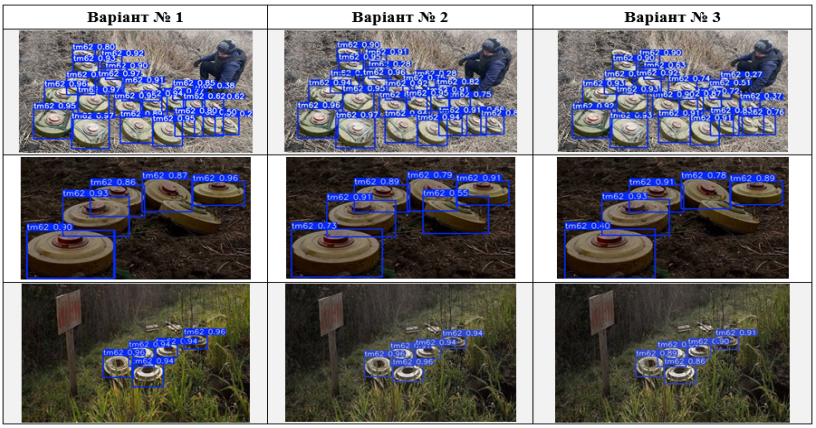


Рисунок 2. – Результати виявлення ВНП на реальних зображеннях тестових вибірок

Результати експерименту підтвердили гіпотезу авторів про доцільність застосування мультимодального генеративного ШІ (*на прикладі Gemini 3.1 Flash Image*) для формування гібридних навчальних датасетів у задачах детекції вибухонебезпечних предметів на реальних зображеннях. Обрана модель генеративного ШІ демонструє високу відтворюваність ключових геометричних і візуальних ознак ВНП на згенерованих зображеннях, що підтверджується отриманими метриками при перевірці навчених CV-моделей YOLOv26m на тестових вибірках. Використання генерованих зображень помітно зменшує (*проте не вирішує повністю*) проблему переносу знань у реальний домен. Пошук оптимальної частки генерованих зображень ВНП у вибірках та аналіз впливу синтетичної складової на ефективність навчання CV-моделей потребує подальших емпіричних досліджень у реальних (*або наближених*) умовах.

У ході здійсненого дослідження було розроблено та опубліковано гібридний датасет, що складається із 276 синтетичних (генерованих) зображень та 61 реального зображення [7].

Список використаних джерел

1. Technologies for Explosive Object Detection for AI-Based / V. Strutynskyi, et al. *Mechanics and Advanced Technologies*. 2026. Vol. 10, No. 1(108). P. 129–140. DOI: 10.20535/2521-1943.2026.10.1(108).356762

2. Tornero-Soria M., Sánchez-Salmerón A.-J., Vidal E. V. Toward a deeper understanding of YOLO26: Block-level architectural analysis and ablation studies. *Applied Sciences*. 2026. Vol. 16, No. 13. Art. 6758. P. 1–49. DOI: 10.3390/app16136758

3. McMillen J., Yilmaz Y. SegGen: An Unreal Engine 5 pipeline for generating multimodal semantic segmentation datasets. *Sensors*. 2025. Vol. 25, No. 17. Art. 5569. P. 1–20. DOI: 10.3390/s25175569

4. Improving Object Detection by Modifying Synthetic Data with Explainable AI / N. Mital, et al. *arXiv:2412.01477v3*. 2024. URL: https://arxiv.org/html/2412.01477v3 (дата звернення: 12.10.2026).

5. Афонін Ю. С., Яковішин А. Я., Савінов В. Ю. Метод формування комбінованого датасету для виявлення вибухонебезпечних предметів засобами YOLOv26. *Системи обробки інформації*. 2026. № 3(186). С. 17–29. DOI: 10.30748/soi.2026.186.02

6. Nano Banana 2 vs. Competitors: The ultimate AI image generator comparison 2025. *Nano Banana 2*. URL: https://nano-banana2.com/blog/nano-banana-2-vs-competitors (дата звернення: 12.10.2026).

7. Afonin Y., Yakovishyn A. Hybrid Image Dataset for Explosive Object Detection : dataset. *Mendeley Data*. 2026. Ver. 1. DOI: 10.17632/mkmkhbvzdd.1. URL: https://data.mendeley.com/datasets/mkmkhbvzdd/1 (дата звернення: 12.10.2026).

УДК 004.31:634

Єлизавета Бугакова
здобувачка першого (бакалаврського) рівня вищої освіти
спеціальності «Комп'ютерна інженерія»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Катерина Обухова
старший викладач кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

АПАРАТНО-ПРОГРАМНИЙ КОМПЛЕКС ДЛЯ ВИЗНАЧЕННЯ СТИГЛОСТІ ПЛОДІВ НА БАЗІ ESP32

Актуальність теми. Визначення ступеня стиглості плодів є важливим завданням у сільському господарстві, оскільки від нього залежать терміни збирання, зберігання, транспортування та реалізації продукції. Традиційне оцінювання переважно ґрунтується на зовнішньому вигляді та досвіді людини, тому його результати можуть бути суб'єктивними. Використання сенсорів дає змогу отримувати кількісні показники властивостей плоду й застосовувати їх для автоматизованого оцінювання стиглості.

У сучасних дослідженнях для неруйнівного визначення якості фруктів використовують оптичні, спектральні та механічні методи. Зокрема, зміни забарвлення пов'язані з процесами дозрівання, а механічні характеристики можуть відображати зміну твердості тканин. Поєднання кількох типів вимірювань є перспективним підходом, оскільки кожен окремий показник має свої обмеження [1; 2].

Мета та принцип роботи пристрою. Метою роботи є розроблення апаратно-програмного комплексу (АПК) для автоматизованого визначення ступеня стиглості плодів шляхом мультисенсорного аналізу їхнього кольору, маси та механічних властивостей із використанням мікроконтролера ESP32. Об'єктами дослідження обрано банани, яблука й мандарини. Основою запропонованого комплексу є мікроконтролер ESP32, який отримуватиме дані від вимірювальних компонентів та виконуватиме алгоритм визначення ступеня стиглості.

Для аналізу забарвлення плодів передбачається використання датчика кольору, що забезпечуватиме отримання цифрових значень

каналів RGB – червоного, зеленого й синього. За потреби отримані дані можуть бути перетворені до колірних моделей HSV (Hue, Saturation, Value) або HSL (Hue, Saturation, Lightness) для аналізу тону, насиченості та характеристик яскравості. Механічні властивості передбачається оцінювати за допомогою датчика сили, який вимірює силу опору плоду під час контрольованого натискання. За допомогою вагового датчика планується вимірювати масу плоду, яку можна буде використовувати як додаткову ознаку під час мультикритеріального оцінювання стиглості. Алгоритм визначення стиглості наведено на рис. 1.

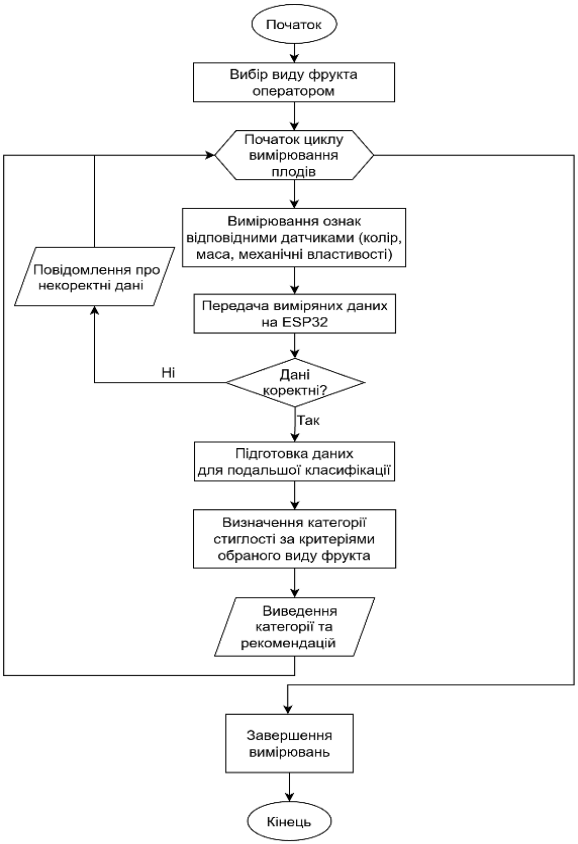


Рисунок 1. – Блок-схема алгоритму визначення ступеня стиглості фруктів

Вид досліджуваного фрукта попередньо обиратиметься оператором. Отримані вимірювальні дані передбачається обробляти на мікроконтролері ESP32 із застосуванням критеріїв оцінювання, визначених для відповідного виду фрукта. Отже, після отримання показників ESP32 виконуватиме алгоритм класифікації, який порівнює виміряні значення з еталонними показниками із датасетів для кожного критерію, визначеними для відповідного виду фрукта. За результатами оцінювання АПК має визначати одну з передбачених категорій: «нестиглий», «стиглий» або «перестиглий» плід. Для забезпечення обґрунтованості результатів класифікації необхідно експериментально встановити межі класифікації. Граничні значення критеріїв планується уточнювати за результатами експериментальних досліджень із використанням власного датасету. Функціональну схему комплексу наведено на рис. 2.

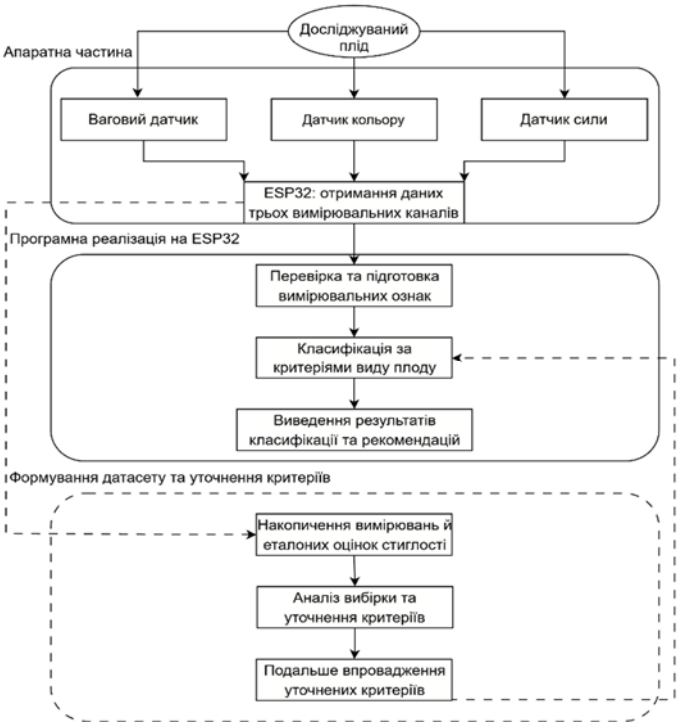


Рисунок 2. – Функціональна схема апаратно-програмного комплексу

Критерії оцінювання та нормативна основа. Під час визначення стиглості важливо враховувати відмінності між видами фруктів. У бананів дозрівання зазвичай супроводжується переходом забарвлення шкірки від зеленого до жовтого, а за подальших змін можуть з'являтися коричневі плями [1; 5]. Для яблук колір оцінюють з урахуванням сорту, оскільки забарвлення плодів може суттєво відрізнятися [3]. Для мандаринів враховують характерне для конкретного виду й сорту забарвлення, однак сам колір не завжди достатній для підтвердження стиглості [4]. У таблиці 1 наведені попередні колірні ознаки для оцінювання ступеня стиглості плодів. Колір шкірки розглядається як один із критеріїв оцінювання стиглості та не забезпечує однозначного визначення її ступеня для всіх досліджуваних плодів.

Таблиця 1. – Попередні колірні ознаки для оцінювання ступеня стиглості плодів

Вид плода	**Категорія стиглості**	**Характерні колірні ознаки**
Банан	Нестиглий	Переважно зелене забарвлення шкірки, можливі жовтуваті ділянки
	Стиглий	Переважно жовта шкірка, можливі залишкові зелені ділянки або поодинокі коричневі цятки
	Перестиглий	Поширення коричневих плям, потемніння значної частини шкірки
Яблуко	Нестиглий	Для частини сортів – інтенсивне зелене основне забарвлення
	Стиглий	Характерне для сорту зелене, жовте, червоне або комбіноване забарвлення
	Перестиглий	Можливе подальше пожовтіння основного забарвлення або локальне потемніння шкірки
Мандарин	Нестиглий	Зелене або зеленувато-жовте забарвлення шкірки
	Стиглий	Характерне для сорту жовто-оранжеве або оранжеве забарвлення, можливі зеленуваті ділянки
	Перестиглий	Не має однозначно визначеного забарвлення; можливі локальні зміни кольору та потемніння

Механічні показники можуть допомагати оцінювати зміни твердості плоду. Для цього вимірюють силу натискання, яка виражається в ньютонах (Н). Отримані значення залежать від місця натискання, площі контакту та способу прикладання навантаження.

Маса фрукта вимірюється в грамах або кілограмах і залежить від сорту та розміру плоду, тому її не можна використовувати як самостійний універсальний показник стиглості [2].

Під час формування критеріїв можна враховувати міжнародні стандарти якості свіжої продукції: UNECE FFV-50 для яблук, UNECE FFV-14 для цитрусових і Codex Alimentarius CXS 205-1997 для бананів [3–5]. Ці документи встановлюють вимоги до якості та стану продукції, але не надають універсальних меж RGB, сили натискання чи маси для автоматичного визначення трьох стадій стиглості. Тому нормативні положення доцільно використовувати як основу для вибору ознак, а числові пороги алгоритму визначати під час експериментальних досліджень. Також передбачається можливість накопичення результатів вимірювань у власному експериментальному датасеті для подальшого аналізу залежності колірних, механічних і вагових показників від ступеня стиглості фруктів.

Висновки. Запропонований апаратно-програмний комплекс на базі ESP32 для визначення ступеня стиглості плодів поєднуватиме три вимірювальні параметри: колір, механічну реакцію та масу плоду. Використання ESP32 забезпечує отримання й оброблення даних від сенсорів для подальшого визначення ступеня стиглості бананів, яблук і мандаринів. Подальші дослідження передбачають вибір конкретних моделей сенсорів, калібрування вимірювальних каналів, формування власного датасету з еталонними оцінками ступеня стиглості плодів та експериментальну перевірку точності алгоритму класифікації.

Список використаних джерел

1. Li B., Lecourt J., Bishop G. Advances in Non-Destructive Early Assessment of Fruit Ripeness towards Defining Optimal Time of Harvest and Yield Prediction – A Review. *Plants*. 2018. Vol. 7, No. 1. Article 3. P. 1–20. DOI: 10.3390/plants7010003

2. Bratu A.-M., Popa C., Bojan M., Logofatu P. C., Petrus M. Non-destructive methods for fruit quality evaluation. *Scientific Reports*. 2021. Vol. 11. Article 7782. P. 1–15. DOI: 10.1038/s41598-021-87530-2

3. UNECE Standard FFV-50. Apples. [Date of issue: Mar 15, 2024]. New York and Geneva : United Nations Economic Commission for Europe, 2023. 17 p. URL: https://unece.org/sites/default/files/2024-03/FFV-50_Apples_2023_e.pdf (дата звернення: 16.10.2026).

4. UNECE Standard FFV-14. Citrus Fruit. [Date of issue: Nov 28, 2025]. New York and Geneva : United Nations Economic Commission for

Europe, 2025. 11 p. URL: https://unece.org/sites/default/files/2025-11/FFV_14_CitrusFruit_E.pdf (дата звернення: 16.10.2026).

5. CXS 205-1997. Standard for Bananas. [Adopted in 1997. Amended in 2005, 2022]. Codex Alimentarius Commission, 2022. 4 p. URL: https://unece.org/fileadmin/DAM/trade/agr/meetings/ge.01/document/Codex%20bananas%20E.pdf (дата звернення: 16.10.2026).

УДК 004.9:656.1

Владислав Воляник
здобувач другого (магістерського) рівня вищої освіти
спеціальності «Комп'ютерна інженерія»
Чорноморський національний університет
імені Петра Могили
Сергій Пузирьов
кандидат фіз.-мат. наук, доцент,
доцент кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

СИСТЕМА ДОПОВНЕНОЇ РЕАЛЬНОСТІ ДОРОЖНЬОГО РУХУ НА БАЗІ RASPBERRY PI ТА OPENCV

Зростання інтенсивності дорожнього руху та значна частка дорожньо-транспортних пригод, спричинених неуважністю водія, стимулюють розвиток систем допомоги водієві (ADAS). Промислові рішення цього класу інтегровано у дорогі автомобілі й вони потребують спеціалізованих обчислювальних модулів, тому для навчальних, дослідницьких і бюджетних застосувань актуальним є створення доступної системи на одноплатному комп'ютері. Технологія доповненої реальності (AR) [1] дає змогу не лише виявити небезпеку, а й показати її водієві безпосередньо на зображенні дороги: накласти контур смуги руху, рамки навколо транспортних засобів, значення відстані та часу до зіткнення. Бібліотека комп'ютерного зору OpenCV [2] містить оптимізовані алгоритми обробки зображень і працює на архітектурі Arm, що робить її природним вибором для платформи Raspberry Pi.

Метою даної роботи є розроблення системи доповненої реальності дорожнього руху на базі Raspberry Pi 5 та OpenCV, яка в режимі реального часу виявляє смугу руху й транспортні засоби перед автомобілем, оцінює відстань до них і виводить графічні підказки поверх відеопотоку з камери.

Структурну схему запропонованої системи наведено на рис. 1. Відеопотік із камери Raspberry Pi Camera Module надходить до одноплатного комп'ютера Raspberry Pi 5 (чотириядерний процесор Arm Cortex-A76 з тактовою частотою 2,4 ГГц). Захоплення кадрів виконується в окремому потоці, що зменшує затримку, пов'язану з очікуванням камери. Після попередньої обробки кадр одночасно потрапляє у два модулі: виявлення маркування смуги та виявлення транспортних засобів. Результати обох модулів об'єднує блок оцінки відстані та часу до зіткнення, а модуль AR формує підсумкове зображення для дисплея або проєкційного індикатора на лобовому склі (HUD).

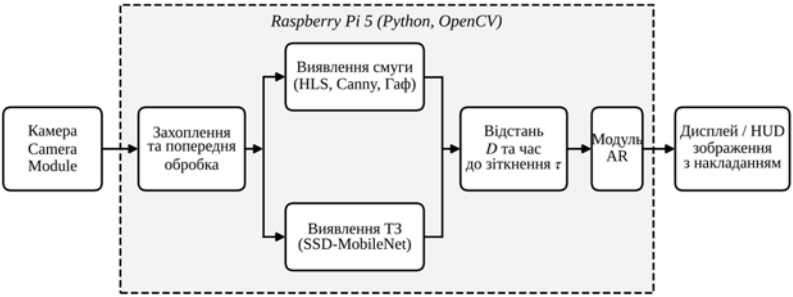


Рисунок 1. – Структурна схема системи доповненої реальності дорожнього руху

Модуль виявлення смуги працює в нижній половині кадру, яка відповідає ділянці дороги перед автомобілем. Зображення переводиться в колірний простір HLS, де за порогами яскравості та колірного тону виділяються білі й жовті лінії маркування, після чого детектор Канні будує карту меж [2]. Відрізки ліній знаходяться ймовірнісним перетворенням Гафа (cv2.HoughLinesP), групуються за знаком нахилу на ліву й праву межі та апроксимуються прямими методом найменших квадратів. Простір між межами заливається напівпрозорим кольором за допомогою cv2.addWeighted, що й створює ефект накладання смуги руху на реальне зображення дороги.

Транспортні засоби виявляються нейромережею SSD [3] з полегшеною основою MobileNet [4], яку завантажують через модуль cv2.dnn. Така комбінація орієнтована на пристрої з обмеженими ресурсами й дає змогу знаходити автомобілі, мотоцикли та вантажівки без графічного прискорювача. Для кожного виявленого об'єкта визначається обмежувальний прямокутник. Відстань до транспортного засобу оцінюється за моделлю камери-обскури [5]:

$$D = \frac{f_y H}{h}, \tag{1}$$

де f_y – фокусна відстань камери в пікселях, визначена калібруванням за шаховою мішенню; H – прийнята висота транспортного засобу (для легкового автомобіля близько 1,5 м); h – висота обмежувального прямокутника в пікселях.

Оскільки висота реального об'єкта відома лише наближено, для оцінки небезпеки використовується не стільки абсолютна відстань, скільки швидкість зростання об'єкта на кадрі. Із пропорційності $h \sim 1/D$ випливає, що час до зіткнення можна знайти без знання швидкості власного автомобіля:

$$\tau = \frac{h_k \Delta t}{h_k - h_{k-1}}, \tag{2}$$

де h_k і h_{k-1} – висоти прямокутника на поточному та попередньому кадрах, а Δt – інтервал між ними. Для зменшення шуму значення τ згладжується ковзним середнім. Рівень підсвічування обирається за величиною τ: понад 4 с – зелений, від 2 до 4 с – жовтий, менше 2 с – червоний, із виведенням попереджувального напису.

Головною вимогою до системи є робота в реальному часі, тому для кожного етапу задано бюджет часу (табл. 1). Частота обробки визначається сумарною тривалістю етапів:

$$F = \frac{1}{\sum_{i=1}^{n} t_i} \geq F_{min}. \tag{3}$$

де t_i – тривалість i-го етапу, n – кількість етапів, $F_{\min}$ – мінімально допустима частота кадрів. За $F_{\min} = 15$ кадр/с сумарний час обробки не повинен перевищувати 66 мс.

Таблиця 1. – Проєктний бюджет часу на обробку одного кадру

№	Етап обробки	Основні засоби OpenCV	t_i, мс
1	Захоплення кадру (окремий потік)	Picamera2, черга кадрів	8
2	Попередня обробка та вибір області інтересу	cv2.cvtColor, cv2.GaussianBlur, ROI	5
3	Виявлення маркування смуги	cv2.inRange, cv2.Canny, cv2.HoughLinesP	20
4	Виявлення транспортних засобів	cv2.dnn.readNet, net.forward	25
5	Оцінка відстані та часу до зіткнення	за формулами (1), (2)	2
6	Побудова AR-накладання	cv2.addWeighted, cv2.rectangle, cv2.putText	6
Разом $\sum t_i$			**66**

Значення в табл. 1 є проєктними орієнтирами й підлягають уточненню за результатами вимірювань на реальних відеозаписах. Найбільш ресурсоємними є виявлення транспортних засобів і маркування, тому саме на них спрямовано оптимізацію: зменшення вхідної роздільної здатності мережі, виконання нейромережевого виведення не на кожному кадрі з відстеженням об’єктів у проміжних кадрах, а також перенесення обчислень на апаратний прискорювач.

Запропоновано архітектуру та алгоритмічне забезпечення системи доповненої реальності дорожнього руху на базі Raspberry Pi 5 та OpenCV, що поєднує виявлення смуги руху, розпізнавання транспортних засобів, оцінку відстані та часу до зіткнення й візуалізацію підказок поверх відеопотоку. Подальшу роботу планується спрямувати на експериментальне визначення частоти кадрів і похибки оцінки відстані в різних умовах освітлення, а також на порівняння варіантів прискорення нейромережі на платформі Raspberry Pi.

Список використаних джерел

1. Kettle L., Lee Y.-C. Augmented Reality for Vehicle-Driver Communication: A Systematic Review. *Safety.* 2022. Vol. 8, No. 4. Art. 84. DOI: 10.3390/safety8040084
2. OpenCV – Open Source Computer Vision Library : documentation. URL: https://docs.opencv.org/4.x/ (дата звернення: 17.10.2026).
3. Liu W., Anguelov D., Erhan D. et al. SSD: Single Shot MultiBox Detector. arXiv:1512.02325. 2015. URL: https://arxiv.org/abs/1512.02325
4. Howard A. G., Zhu M., Chen B. et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. arXiv:1704.04861. 2017. URL: https://arxiv.org/abs/1704.04861
5. Lee S., Han K., Park S., Yang X. Vehicle Distance Estimation from a Monocular Camera for Advanced Driver Assistance Systems. *Symmetry.* 2022. Vol. 14, No. 12. Art. 2657. DOI: 10.3390/sym14122657

УДК 004.93:004.382.7

Станіслав Гануліч
здобувач другого (магістерського) рівня вищої освіти
спеціальності «Комп'ютерна інженерія»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Сергій Пузирьов
кандидат фіз.-мат. наук, доцент,
доцент кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

МЕТОДИ ТА ЗАСОБИ КОМП'ЮТЕРНОГО ЗОРУ НА БАЗІ RASPBERRY PI

Автоматизований облік присутності працівників в офісі потребує своєчасної реєстрації входу й виходу та передавання повідомлень відповідним підрозділам. Окремим завданням є виявлення осіб із блеклиста – переліку небажаних осіб, поява яких має спричиняти сповіщення служби охорони. Для цих сценаріїв пропонується локальна

система комп'ютерного зору на базі одноплатного комп'ютера Raspberry Pi. Оброблення відео безпосередньо на пристрої дає змогу передавати до зовнішніх сервісів лише сформовані події, а не безперервний відеопотік.

Метою даної роботи є обґрунтування методів та архітектури програмно-апаратної системи на базі Raspberry Pi й OpenCV для розпізнавання працівників компанії, реєстрації їхньої присутності та автоматичного інформування департаментів і охорони. Для досягнення мети передбачено формування датасету облич, вибір засобів виявлення й розпізнавання, визначення напрямку проходу та розроблення правил надсилання повідомлень. Основними сценаріями є поява співробітника в офісі, його вихід і виявлення людини з блек-листа.

Апаратну основу прототипу пропонується побудувати на Raspberry Pi 5, який має інтерфейси для підключення камер і мережевого обміну [1]. До системи також входять камера або камери для спостереження за проходом, локальна база еталонів і модуль сповіщень (рис. 1). Камери розміщуються так, щоб обличчя були видимі під час руху в обох напрямках. Для отримання кадрів із сумісної камери Raspberry Pi можна використовувати бібліотеку Picamera2 [2]. Остаточні параметри відеопотоку та конфігурацію плати необхідно визначити під час випробувань.

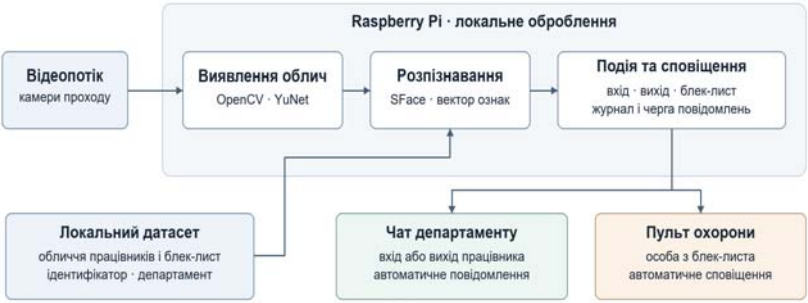


Рисунок 1. – Структура системи розпізнавання та сповіщення

Програмне оброблення кадру передбачає виявлення обличчя, визначення опорних точок, вирівнювання зображення та обчислення вектору ознак. Для реалізації пропонується застосувати інтерфейси FaceDetectorYN і FaceRecognizerSF бібліотеки OpenCV [3]. Детектор YuNet обрано як компактну модель, орієнтовану на периферійні пристрої з обмеженими обчислювальними ресурсами [4]. Для

отримання вектору ознак використовується попередньо навчена модель SFace, підхід до навчання якої регулює внутрішньокласові й міжкласові відстані [5]. Таке розділення дає змогу окремо налаштовувати виявлення облич та встановлення особи.

Датасет формується з дозволених до використання зображень працівників компанії та осіб із блек-листа. Для кожної людини передбачається серія знімків за різного освітлення, ракурсу й відстані до камери. Перед включенням до еталонної бази необхідно відбраковувати розмиті, надто темні та неправильно підписані зображення. Знімки з однією особою об'єднуються під сталим ідентифікатором. Після вирівнювання облич для них обчислюються вектори ознак; саме ці вектори використовуються для порівняння під час роботи системи.

Запис працівника має містити його ідентифікатор, ім'я, належність до департаменту та ідентифікатор відповідного службового чату. Запис людини з блек-листа містить окрему ознаку, за якою система обирає маршрут до охорони. Додавання нової особи полягає в обчисленні та збереженні її еталонів і не потребує повторного навчання всієї нейронної мережі. Оновлення набору має також передбачати вилучення застарілих записів і зміну департаменту працівника. Редагування списків та доступ до зображень надаються лише уповноваженому адміністратору.

Під час ідентифікації вектор поточного обличчя порівнюється з локальними еталонами за косинусною подібністю. Рішення про збіг приймається за перевищення налаштованого порога та достатнього відриву від другого кандидата. Пороги визначаються на валідаційній вибірці з урахуванням умов конкретного проходу. Невпевнений результат позначається як «невідома особа» і не прирівнюється до збігу з блек-листом. Для зменшення випадкових помилок особа має підтверджуватися в кількох послідовних кадрах у межах однієї траєкторії руху. Реєстрація входу та виходу потребує аналізу руху відносно двох контрольних зон по різні боки проходу. Перехід із зовнішньої зони у внутрішню відповідає входу, а зворотний – виходу. Ідентифікатор розпізнаного обличчя зв'язується з відстежуваною траєкторією та зберігається до завершення проходу. Зникнення обличчя з кадру саме по собі не означає, що працівник залишив офіс. За втрати траєкторії або неоднозначного напрямку подія не підтверджується автоматично, оскільки це може спричинити помилкову зміну стану присутності.

Для кожного працівника передбачено стани «в офісі», «поза офісом» і «невідомо». Підтверджений вхід переводить запис у стан «в

офісі», а підтверджений вихід – у стан «поза офісом». Якщо початкових даних немає або після збою неможливо встановити актуальну присутність, використовується стан «невідомо». Повторне розпізнавання людини, яка стоїть у проході, не повинно створювати нової події входу. Для кількох людей одночасно необхідно підтримувати окремі траєкторії та не переносити ідентифікатор між ними.

Після підтвердженого входу система автоматично надсилає до чату департаменту повідомлення «Працівник [ім’я] в офісі» із часом події. Підтверджений вихід формує повідомлення «Працівник [ім’я] залишив офіс» у той самий чат. Відповідність між працівником і чатом визначається за локальною базою. Надсилання пропонується реалізувати через програмний інтерфейс обраного корпоративного месенджера; конкретний сервіс не впливає на алгоритм розпізнавання. Окремий модуль інтеграції дає змогу змінити месенджер без зміни алгоритмів аналізу відео. Поява людини з блек-листа є самостійною подією. Після підтвердженого збігу система автоматично передає на пульт охорони сповіщення з часом, місцем спостереження та ідентифікатором знайденого запису. Для цього не потрібно чекати завершення входу або виходу: достатньо появи обличчя в контрольованій зоні та підтвердження ідентифікації. Подальшу перевірку виконує оператор охорони. Повторні кадри одного спостереження об’єднуються в одну подію, щоб не створювати потік однакових тривожних повідомлень.

Модуль сповіщень відокремлюється від оброблення відео за допомогою черги подій. Кожна подія містить унікальний ідентифікатор, час, тип, ідентифікатор особи та одержувача. У разі втрати мережевого з’єднання повідомлення зберігається локально до повторного надсилання. Після відновлення зв’язку час виявлення та час доставки обліковуються окремо, щоб затримане повідомлення не сприймалося як поточна поява людини. Стан доставки фіксується в журналі, а ідентифікатор події використовується для усунення повторів на стороні приймача. Робота на Raspberry Pi потребує узгодження роздільності кадру, частоти його оброблення та кількості одночасно видимих облич. Надмірне зменшення зображення може спричинити втрату деталей, потрібних для ідентифікації, тоді як великий кадр збільшує обсяг обчислень. Тому ці параметри пропонується підбирати спільно за якістю розпізнавання та затримкою. За перевантаження слід уникати накопичення застарілих кадрів у черзі й обробляти актуальні дані. Мережеве надсилання має

виконуватися незалежно, щоб недоступність месенджера не зупиняла аналіз відео.

Для перевірки якості необхідно відокремити еталонний набір від валідаційних і тестових записів. Відео для різних вибірок отримуються під час окремих сеансів знімання; сусідні кадри одного проходу не слід розподіляти між ними. Тестовий набір має містити відомих працівників, осіб із блек-листа та людей, відсутніх в еталонній базі. Такий склад дає змогу оцінити не лише правильні розпізнавання, а й помилкове приписування невідомій людині ідентифікатора працівника або запису з блек-листа.

Оцінювання передбачає вимірювання часток хибних збігів і пропусків розпізнавання, точності визначення напрямку проходу та затримки від появи людини до формування події. Окремо визначаються час доставки сповіщення, кількість дубльованих повідомлень, завантаження процесора й використання пам'яті. Сценарії мають охоплювати зміну освітлення, часткове закриття обличчя, зупинку в проході, одночасний рух кількох людей і тимчасову втрату зв'язку. Для кожного проходу вручну фіксуються очікувані особа, напрямок і одержувач, після чого вони зіставляються з журналом системи.

Таким чином, запропоновано архітектуру системи комп'ютерного зору на базі Raspberry Pi, що поєднує формування датасету облич, розпізнавання засобами OpenCV та адресне інформування. Реєстрація входу й виходу працівника пов'язана зі станом присутності та чатом його департаменту, а виявлення людини з блек-листа – з окремим автоматичним сповіщенням охорони. Подальша робота передбачає створення прототипу й експериментальне визначення параметрів, за яких забезпечується прийнятна якість ідентифікації та своєчасність повідомлень.

Список використаних джерел

1. Raspberry Pi. Raspberry Pi 5. URL: https://www.raspberrypi.com/products/ raspberry-pi-5/ (дата звернення: 14.10.2026).

2. Raspberry Pi. Camera software. Raspberry Pi Documentation. URL: https://www.raspberrypi.com/documentation/computers/camera_software.html (дата звернення: 14.10.2026).

3. OpenCV. DNN-based Face Detection And Recognition. OpenCV Documentation. URL: https://docs.opencv.org/4.x/d0/dd4/tutorial_dnn_face. html (дата звернення: 14.10.2026).

4. Wu W., Peng H., Yu S. YuNet: A Tiny Millisecond-level Face Detector. Machine Intelligence Research. 2023. Vol. 20. P. 656–665. DOI: 10.1007/s11633-023-1423-y

5. Zhong Y., Deng W., Hu J., Zhao D., Li X., Wen D. SFace: Sigmoid-Constrained Hypersphere Loss for Robust Face Recognition. IEEE Transactions on Image Processing. 2021. Vol. 30. P. 2587–2598. DOI: 10.1109/TIP.2020.3048632

УДК 004.3:681.5

Ілля Гуляєв
здобувач другого (магістерського) рівня вищої освіти
спеціальності «Комп'ютерна інженерія»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євген Дарнапук
PhD, доцент,
в.о. завідувача кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ EDGE-AI КЕРУВАННЯ МОБІЛЬНОЮ ПЛАТФОРМОЮ НА БАЗІ NVIDIA JETSON NANO

Актуальність теми. Бортова система керування мобільною платформою має обробити зображення та своєчасно передати команду приводам. Її затримка залежить від одержання кадру, підготовки даних, нейромережевого виведення й роботи інтерфейсів. На NVIDIA Jetson Nano ці етапи виконуються за обмежених обчислювальних, енергетичних і теплових ресурсів. Середня частота обробки кадрів може приховувати тривалі цикли без оновлення команд.

DeepPicar продемонстрував вплив конкуренції за спільні ресурси на час виконання вбудованого нейромережевого керування [1]. Для Jetson Nano доцільно визначити склад стенда й умови його перевірки під навантаженням.

Мета даної роботи – обґрунтувати апаратну структуру системи Edge-AI керування на NVIDIA Jetson Nano та визначити параметри

живлення, охолодження й використання ресурсів для відтворюваного оцінювання своєчасності команд.

Обчислювальна платформа. Розглядається Jetson Nano з чотириядерним ARM Cortex-A57, GPU Maxwell зі 128 ядрами CUDA та 4 ГБ LPDDR4 із пропускною здатністю 25,6 ГБ/с [2]. CPU виконує керувальну логіку, обмін із периферією та підготовку даних; GPU може виконувати нейромережеві обчислення. Скорочення всього циклу залежить від розподілу операцій і реалізації моделі.

CPU та GPU використовують спільну оперативну пам'ять. Модель, кадри й системні процеси враховують у межах єдиного ресурсу. Для запобігання затримкам контролюють доступну пам'ять, підкачування та накопичення кадрів у черзі.

Структура апаратного стенда. Запропоновано стенд із камерою, платою з модулем Jetson Nano, зовнішнім контролером приводів, джерелом живлення та охолодженням. Для камери використовують MIPI CSI-2 або USB, для контролера – підтримуваний платою інтерфейс, наприклад I²C чи UART [2]. Конкретне підключення залежить від плати-носія та периферії. Проєктну структуру наведено на рисунку 1; пунктир позначає збирання даних.

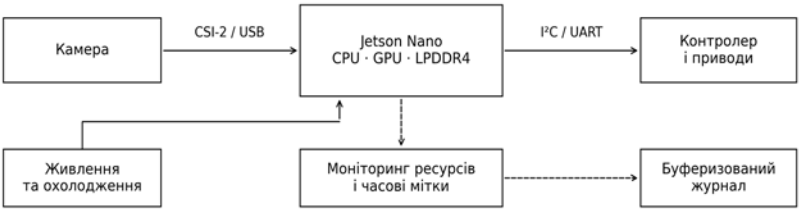


Рисунок 1. – Запропонована структура стенда на базі Jetson Nano

Jetson Nano формує команди, а зовнішній контролер передає їх приводам. Для стенда доцільно визначити реакцію на втрату чи прострочення команди, наприклад зупинку після тайм-ауту. Це вимога до проєкту, яку потрібно перевіряти окремо від швидкодії нейромережі.

Живлення й тепловий режим. У документації NVIDIA наведено режими 5W і MAXN із бюджетом 10 Вт. Вони задають різні обмеження на частоту CPU/GPU та кількість активних процесорних ядер [3]. Назва режиму не є вимірюванням споживання всього робота. Камера, плата-носій, вентилятор, перетворювачі й приводи формують

додаткове навантаження. Для порівняння потрібні однакові умови живлення та реєстрація обраного режиму.

Для стенда пропонується розділити живлення обчислювальної та силової частин, узгодивши сигнальні рівні й опорну землю неізольованих інтерфейсів. Слід врахувати пускові струми приводів і втрати в кабелях. Напругу та струм вимірюють у визначеній точці; для оцінки енергії враховують часову роздільну здатність вимірювача.

Нагрівання може спричиняти обмеження тактових частот; Jetson Nano має програмні й апаратні механізми такого регулювання [3]. Тому разом із температурами зберігають частоти, умови обдування та часові показники. Повторні запуски починаються за зіставного теплового стану. Причини затримок після зміни охолодження перевіряють за допомогою контрольованого експерименту.

Телеметрія апаратного стану. Для системних відліків придатна утиліта tegrastats: вона відображає використання RAM і SWAP, навантаження та частоти CPU/GPU, стан контролера пам'яті й температурні канали [4]. Відсоток навантаження слід інтерпретувати разом із частотою відповідного вузла. Доступність окремих полів перевіряють на конкретному пристрої; відсутні відліки не замінюють нульовими. Узагальнені апаратні чинники наведено в таблиці 1.

Таблиця 1. – Апаратні чинники та їх урахування під час випробувань

Чинник	**Що реєструвати**	**Для чого враховувати**
Обчислювальний режим	Режим nvpmodel, частоти й активні ядра	Розмежувати ефект налаштувань CPU/GPU та моделі
Живлення	Точку вимірювання, напругу, струм	Відрізнити споживання плати від витрат усієї платформи
Охолодження	Температури, частоти, стан вентилятора	Перевірити зміну затримок за тривалого навантаження
Пам'ять і введення	RAM/SWAP, чергу кадрів, втрати кадрів	Виявити нестачу ресурсів і використання застарілих даних
Вихідний інтерфейс	Мітки команд, підтвердження за наявності	Розрізнити передавання команди та її виконання

Частоту моніторингу задають окремо від частоти керування. Події й відліки реєструють на спільній монотонній часовій шкалі, а потім пов’язують за оголошеним правилом і допустимою часовою відстанню. Повільний монітор не забезпечує вимірювання для кожного кадру й не відновлює піків між відліками. Для з’ясування причин затримок може знадобитися додаткове профілювання.

Порядок апаратного оцінювання. Режим nvpmodel або охолодження змінюють за незмінних моделі, розміру зображення, контролера та сценарію руху. Для запуску фіксують версії програмного середовища, початкову температуру, живлення й параметри камери. Після прогрівання виконують повторні запуски та чергують порядок конфігурацій, щоб нагрівання або розряджання акумулятора не збігалися з одним варіантом.

Для запропонованого стенда часовим показником обрано інтервал від доступності кадру застосунку до подання відповідної команди на вихідний інтерфейс. Експозиція до початкової мітки й механічна реакція приводів потребують окремих вимірювань. Разом із середнім часом доцільно визначати 95-й процентиль, частку перевищень дедлайну та найдовшу послідовність перевищень: значення послідовних пропусків для стійкості контуру керування досліджено в [5]. Дедлайн встановлюють відповідно до динаміки об’єкта.

Потрібно також порівняти запуски з моніторингом і без нього, оскільки журналювання створює додаткове навантаження. Час збирання та збереження даних слід враховувати під час вибору частоти телеметрії.

Спочатку перевіряють дотримання часових, ресурсних і теплових обмежень. Прийнятні конфігурації порівнюють за окремими критеріями, зокрема затримкою та виміряним енергоспоживанням. Вибір залежить від вимог до конкретної платформи. Без незалежного вимірювання траєкторії не можна зробити висновків про точність руху.

Висновки. Обґрунтовано структуру стенда на NVIDIA Jetson Nano та умови його апаратного оцінювання. Живлення, охолодження й використання обчислювальних ресурсів потрібно розглядати разом із часом формування команд. Запропоновані умови випробувань дають основу для перевірки компромісу між своєчасністю команд і витратами ресурсів. Характеристики фізичної установки мають бути встановлені експериментально.

Список використаних джерел

1. Bechtel M. G., McEllhiney E., Kim M., Yun H. DeepPicar: A Low-Cost Deep Neural Network-Based Autonomous Car. 2018 IEEE 24th International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA). 2018. P. 11–21. DOI: 10.1109/RTCSA.2018.00011

2. Jetson Nano. NVIDIA Developer. URL: https://developer.nvidia.com/embedded/jetson-nano (дата звернення: 19.10.2026).

3. Power Management for Jetson Nano and Jetson TX1 Devices. NVIDIA Jetson Linux Developer Guide. Release 32.7.6. URL: https://docs.nvidia.com/jetson/archives/l4t-archived/l4t-3276/Tegra%20Linux%20Driver%20Package%20Development%20Guide/power_management_nano.html (дата звернення: 19.10.2026).

4. tegrastats Utility. NVIDIA Jetson Linux Developer Guide. Release 32.7.6. URL: https://docs.nvidia.com/jetson/archives/l4t-archived/l4t-3276/Tegra%20Linux%20Driver%20Package%20Development%20Guide/AppendixTegraStats.html (дата звернення: 19.10.2026).

5. Maggio M., Hamann A., Mayer-John E., Ziegenbein D. Control-System Stability Under Consecutive Deadline Misses Constraints. 32nd Euromicro Conference on Real-Time Systems (ECRTS 2020). 2020. Vol. 165. P. 21:1–21:24. DOI: 10.4230/LIPIcs.ECRTS.2020.21

УДК 004.932:007.52

Нікіта Гюльмамедов
здобувач другого (магістерського) рівня вищої освіти
спеціальності «Комп'ютерна інженерія»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Іван Бурлаченко
старший викладач кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

СЕГМЕНТАЦІЯ КОЛЬОРІВ НА БАЗІ КОМПОНЕНТІВ МУЛЬТИАГЕНТНИХКІБЕРФІЗИЧНИХ СИСТЕМ

Процес автоматичного виявлення, ідентифікації та аналізу певних кольорів або їхніх відтінків на цифровому зображенні чи у кадрі відеопотоку за допомогою методів комп'ютерного зору набуває дедалі більшого поширення. У багатьох сферах застосовується сегментація зображення на основі хроматичних характеристик пікселів для виявлення об'єктів із заданими колірними параметрами. Процес базується на аналізі значень у різних колірних просторах та дозволяє відокремити цільовий об'єкт від фону.

Платформи Raspberry Pi та ESP32-CAM інтегруються у загальний клас кіберфізичних систем [1] і виступають базовими інструментами для побудови вузлів мультиагентних систем (MAS), Інтернету речей (IoT), а також бездротових мультимедійних сенсорних мереж (WMSN). Незважаючи на функціональну подібність у розв'язанні задач обробки медіаданих та бездротової передачі інформації, зазначені платформи репрезентують фундаментально різні класи обчислювальних архітектур за своїми структурно-функціональними характеристиками [2].

Апаратна специфікація Raspberry Pi визначає цей пристрій як одноплатний комп'ютер, побудований на базі повноцінної системи на кристалі з процесором прикладного рівня архітектури ARM Cortex-A. Функціонування платформи забезпечується операційними системами загального призначення з розподілом часу на базі ядра Linux, які здійснюють динамічне керування віртуальною пам'яттю. Завдяки багатоядерній архітектурі та значному обсягу оперативної пам'яті вимірюваному гігабайтами, Raspberry Pi реалізує парадигму

високопродуктивних паралельних обчислень з підтримкою складних конвеєрів штучного інтелекту, що зумовлює відносно високий рівень енергоспоживання (рис. 1) у кілька ват і потребує застосування систем охолодження.

На противагу цьому, плата розробки ESP32-CAM базується на архітектурі мікроконтролера типу Xtensa LX6 або аналогічних RISC-рішеннях. Програмне забезпечення виконується безпосередньо на апаратній логіці за принципом bare-metal або під керуванням операційних систем реального часу, оперуючи обмеженим обсягом інтегрованої пам'яті на рівні кількох мегабайт. Обчислювальна парадигма мікроконтролера орієнтована на енергоефективні та подієво-орієнтовані процеси в межах вбудованих систем. Це дозволяє досягти ультранизького енергоспоживання у діапазоні міліват із можливістю переходу в глибокі режими сну, що є критично важливим для автономних сенсорних вузлів.

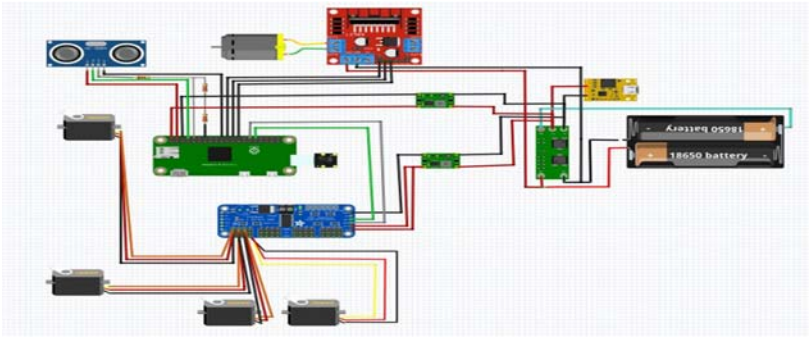


Рисунок 1. – Компонентна схема мультиагентної кіберфізичної системи

У межах ієрархічної структури граничних обчислень (Edge Computing) досліджувані платформи займають різні рівні абстракції та топологічного розміщення. Мікроконтролер ESP32-CAM класифікується як інтелектуальний сенсорний вузол або пристрій дальньої границі (Far-Edge), функціональне призначення якого полягає в детектуванні первинного відеопотоку та здійсненні первинної низькорівневої фільтрації даних. Одноплатний комп'ютер Raspberry Pi натомість позиціонується як граничний шлюз або локальний мікросервер ближньої границі (Near-Edge), спроможний агрегувати інформаційні потоки від множини периферійних вузлів, здійснювати їх комплексний аналіз за допомогою ресурсомістких штучних

нейронних мереж та виконувати функції координації гетерогенної мережі.

З позиції цифрової обробки сигналів обидва апаратні рішення є невід'ємними компонентами вбудованих систем комп'ютерного зору, що функціонують у межах концепції наближення обчислювальних потужностей до джерела генерації даних (Data-Centric Computing), мінімізуючи затримки передачі сигналів та оптимізуючи пропускну здатність каналів зв'язку. У загальній архітектурі Інтернету речей ці платформи розглядаються як гетерогенне середовище для дослідження та розгортання моделей периферійного штучного інтелекту (Edge AI), причому ESP32-CAM є специфічним об'єктом для застосування методів TinyML у мобільних системах з обмеженими апаратними ресурсами.

Метод Оцу [3] використовується в комп'ютерному зорі та обробці зображень для автоматичного визначення порогового значення зображення. Головна мета методу визначити оптимальне порогове значення, яке ефективно розділяє пікселі зображення на два окремі класи: передній план і фон.

Алгоритм працює, використовуючи гістограму зображення (рис. 2), яка представляє розподіл інтенсивності пікселів. Метод Оцу спрямований на пошук порогового значення, яке мінімізує дисперсію в межах кожного класу (внутрішньокласова дисперсія) або, що еквівалентно, максимізує дисперсію між класами (міжкласова дисперсія). Досягаючи цього, він визначає поріг, який найкраще розрізняє області переднього та заднього плану, що робить його корисним для завдань сегментації зображень.

Після визначення порогу всі пікселі з інтенсивністю нижче порогу відносяться до класу фону, тоді як ті, що вище порогу, відносяться до класу переднього плану. Цей процес встановлення порогу дозволяє виділити об'єкти або області інтересу із зображення.

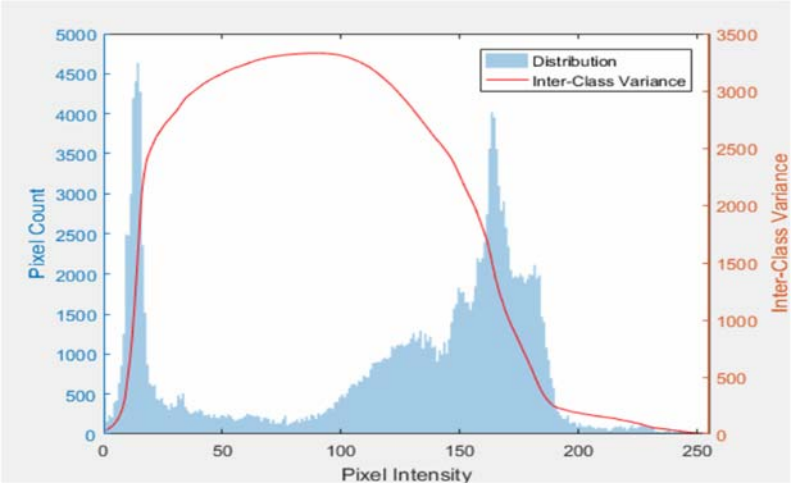


Рисунок 2. – Результати сегментації кольорів мультиагентною кіберфізичною системою

На рис. 2 зображено графік, що ілюструє принцип роботи методу Оцу для автоматичного вибору порогу бінаризації та сегментації зображень. Графік поєднує в собі гістограму розподілу інтенсивності пікселів та криву міжкласової дисперсії, що дозволяє візуально оцінити процес розділення зображення на два класи – передній план та фон. Блакитна зафарбована область відповідає гістограмі розподілу інтенсивності пікселів (Distribution). Вісь X відображає рівні інтенсивності пікселів від 0 до 250, а ліва блакитна вісь Y показує їхню кількість (Pixel Count). На гістограмі чітко спостерігається бімодальний розподіл із двома вираженими піками: перший знаходиться в області темних відтінків (інтенсивність близько 15), а другий – у світліших тонах (інтенсивність близько 165). Червона лінія демонструє міжкласову дисперсію (Inter-Class Variance), значення якої відкладаються по правій помаранчевій осі Y. Головна мета методу Оцу – знайти таке значення порогу, за якого міжкласова дисперсія досягає свого максимуму. На цьому зображенні пік червоної кривої припадає на діапазон інтенсивності приблизно 90–100 із значенням дисперсії біля 3350, що означає, що саме це значення є оптимальним порогом для сегментації об’єктів від фону.

Перевагою методу Оцу є те, що він забезпечує автоматизований та керований даними підхід до вибору порогу, усуваючи необхідність ручного втручання. Даний метод є ефективним під час аналізу

зображень компонентами мультиагентних кіберфізичних систем, включаючи розпізнавання об'єктів, виявлення країв та сегментацію зображень.

Список використаних джерел

1. Challenges in the development of cyber-physical systems–insights from research and industrial practice / L. Gesmann, M. Fischer, A. Lauff, et al. Procedia Computer Science. 2026. Vol. 277. P. 1963–1972.
2. Kanwal Q., Kumar D., Halder N., Al-Ghamdi S. G. Cyber-physical security and sustainability: Foundations for a resilient industry. Industry 4.0 and Sustainability / S. G. Al-Ghamdi (ed.). [Sl.] : Elsevier, 2026. P. 121–139.
3. Performance analysis of image thresholding: Otsu technique / T. Y. Goh, S. N. Basah, H. Yazid, et al. Measurement. 2018. Vol. 114. P. 298–307.

УДК 004.383:004.7

Юрій Івченко
здобувач першого (бакалаврського) рівня вищої освіти
спеціальності «Комп'ютерна інженерія»
Борис Салтовський
старший викладач кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ПРОГРАМНО-АПАРАТНИЙ КОМПЛЕКС ДЛЯ ЛОКАЛЬНОГО МЕТЕОРОЛОГІЧНОГО МОНІТОРИНГУ

Актуальність теми. Розвиток технологій Інтернету речей створює можливості для розробки компактних пристроїв моніторингу навколишнього середовища. Такі пристрої дають змогу отримувати локальні метеорологічні дані безпосередньо за місцем проживання, що може бути корисним для власників приватних будинків і міських квартир.

Одним із практичних рішень є створення модуля IoT-метеостанції, яку можна розмістити на підвіконні або на прибудинковій території для вимірювання погодних параметрів.

Мета даної роботи – розробити модуль метеостанції для вимірювання та моніторингу параметрів навколишнього середовища за допомогою комплексу датчиків і мікроконтролера ESP32 [1] з передачею даних на локальний вебсервер.

Основні компоненти системи. Пристрій побудовано на базі мікроконтролера ESP32 та комплексу датчиків параметрів навколишнього середовища. Для вимірювання температури й вологості повітря застосовано цифровий датчик AM2320, який вирізняється високою точністю заводського калібрування та підтримкою інтерфейсу I2C, що забезпечує раціональне використання апаратних ресурсів мікроконтролера. Як датчик атмосферного тиску інтегровано цифровий модуль BMP280, перевагою якого є низьке енергоспоживання, що критично важливо для портативних пристроїв. Для реєстрації рівня ультрафіолетового випромінювання використано високоточний і компактний датчик VEML6075 [2].

Принцип роботи. Після ввімкнення пристрою мікроконтролер ESP32 з періодичністю в 2 хвилини здійснює зчитування даних із цифрових датчиків BMP280, AM2320 та VEML6075. Отримані параметри навколишнього середовища обробляються мікроконтролером, формуються в пакет у форматі JSON та надсилаються за допомогою протоколу MQTT [3] мережею Wi-Fi на локальний сервер. Запущений на сервері вебзастосунок приймає дані для їх подальшого відображення та аналізу. Алгоритм роботи наведено на рис. 1.



Рисунок 1. – Алгоритм роботи пристрою

Практичне застосування. Розроблений пристрій може бути використаний у різних сферах для локального моніторингу навколишнього середовища: у приватному домогосподарстві – для встановлення на ґанку чи прибудинковій території з метою контролю актуальних погодних умов; у житлових квартирах – для розміщення на підвіконні з метою оперативного відстеження мікроклімату; у сфері «розумного дому» – як надійне джерело метеорологічних даних для інтеграції в локальні системи автоматизації.

Переваги розробки. Основними перевагами запропонованого рішення є: низька вартість компонентів, простота монтажу й

експлуатації, точність вимірювання базових метеорологічних показників, а також гнучкість у програмуванні та обробці даних за допомогою мікроконтролера ESP32.

Крім того, система вирізняється компактністю, автономністю та можливістю зручного розміщення на підвіконні або прибудинковій території для оперативного моніторингу навколишнього середовища.

Висновки. Розроблений пристрій на базі мікроконтролера ESP32 та спеціалізованих датчиків (AM2320, BMP280, VEML6075) підтвердив ефективність локального метеорологічного моніторингу як перспективного напрямку розвитку систем Інтернету речей. Отримані результати демонструють можливість практичного застосування системи у приватному домогосподарстві, житлових квартирах та у сфері «розумного дому» для забезпечення оперативного контролю погодних умов і підвищення комфорту користувачів.

Список використаних джерел

1. Espressif Systems. ESP32 Series Datasheet. 2024. 56 с. URL: https://www.espressif.com (дата звернення: 11.10.2026).

2. Adafruit VEML6075 UVA UVB and UV index sensor breakout. URL: https://www.adafruit.com/product/3964 (дата звернення: 11.10.2026).

3. Доля А. С. Порівняльний аналіз інтернет-протоколів MQTT та CoAP для використання в інтернеті речей. *Радіоелектроніка та молодь у XXI столітті* : матеріали 27-го Міжнар. молодіж. форуму, 10–12 трав. 2023 р. Харків : ХНУРЕ, 2023. Т. 1. С. 31–32.

UDC 621.396.67:534.87

Yaroslav Krainyk
Cand. Tech. Sc., Associate Professor
Petro Mohyla Black Sea National University, Mykolaiv, Ukraine
Kyiv School of Economics
Kyiv, Ukraine
Andrii Kuchynskyi
Master's student
Kyiv School of Economics
Kyiv, Ukraine

NARROWBAND AUDIO SIGNAL DETECTION USING UNIFORM CIRCULAR ARRAY

Spatial spectrum estimation and Direction of Arrival (DOA) detection play a fundamental role in acoustic signal processing, radar, and wireless communications. While linear array topologies like the Uniform Linear Array (ULA) suffer from 180° front-back directional ambiguity and limited spatial coverage, planar two-dimensional configurations provide comprehensive azimuth discrimination. Among these, the Uniform Circular Array (UCA) offers an optimal symmetrical geometry capable of full 360° seamless azimuth source tracking without directional blind spots.

In this work, a 6-element UCA operating on the Z = 0 plane is modeled in a simulated acoustic room environment (8m x 8m x 3m) using the Phased Array System Toolbox. The array possesses a radius $R = \frac{\lambda}{2}$, where $\lambda = c / f$ denotes the acoustic wavelength for a narrowband single-frequency source operating at $f_0 = 1000Hz$ in free space (c = 343 m/s).

The array response vector (steering vector) for an N-element UCA placed symmetrically around the origin is expressed mathematically as:

$$a(\theta,\varphi) = \begin{bmatrix} e^{jkR\sin\varphi\cos(\theta-\gamma_1)} \\ e^{jkR\sin\varphi\cos(\theta-\gamma_2)} \\ \cdots \\ e^{jkR\sin\varphi\cos(\theta-\gamma_N)} \end{bmatrix} \quad (1)$$

where $k = \frac{2\pi}{\lambda}$ is the wavenumber, $\theta \in [-180°, +180°]$ is the azimuth angle, φ is the elevation angle, and $\gamma_n = \frac{2\pi(n-1)}{N}$ represents the angular position of the n-th microphone element for n=1, 2, …, N.

The spatial covariance matrix R_x of the array observation vector $x(t)$ subject to Additive White Gaussian Noise (AWGN) with SNR = 20 dB is given by

$$R_x = \mathrm{E}[x(t)x^H(t)] = U_s \Sigma_s U_s^H + U_n \Sigma_n U_n^H \tag{2}$$

where U_s spans the signal subspace and U_n spans the noise subspace orthogonal to the incoming signal array steering vector. Applying the 2D Multiple Signal Classification (MUSIC) spatial pseudo-spectrum estimator yields:

$$P_{MUSIC}(\theta, \varphi) = \frac{1}{a^H(\theta, \varphi) U_n U_n^H a(\theta, \varphi)} \tag{3}$$

To evaluate dynamic tracking responsiveness, a moving source trajectory was configured along a circular sweep path from $\theta = 45°$ down to $\theta = 0°$ at a fixed radial distance $R_s = 5m$. Coordinates in the Cartesian system are mapped directly via:

$$\theta_{true}(t) = a \tan 2d(y(t), x(t)) \tag{4}$$

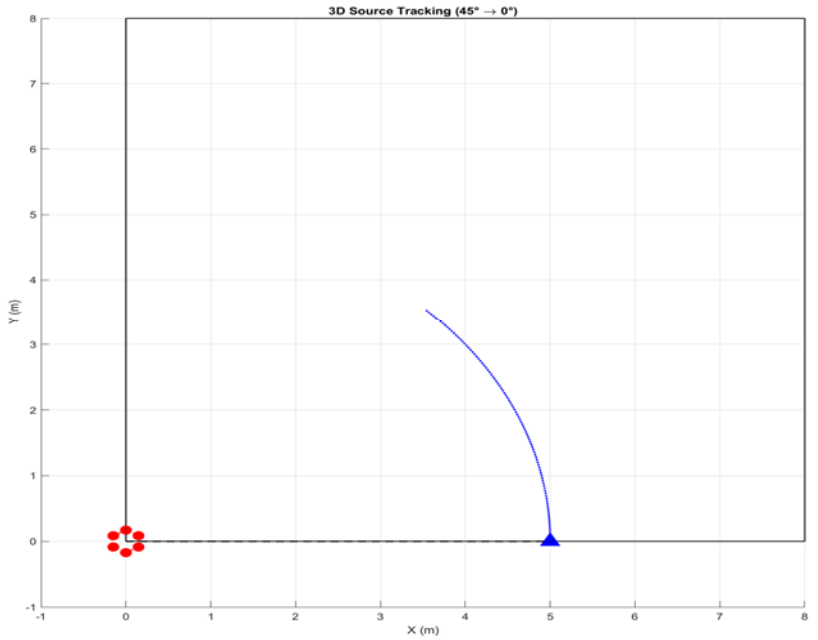


Figure 1. – Top-down view of the simulation setup: 6-element UCA, line of sight, and moving source trajectory

In dynamic real-time frame evaluation, instant covariance state clearing (reset) prevents static memory retention artifacts in snapshot buffers across successive iterations.

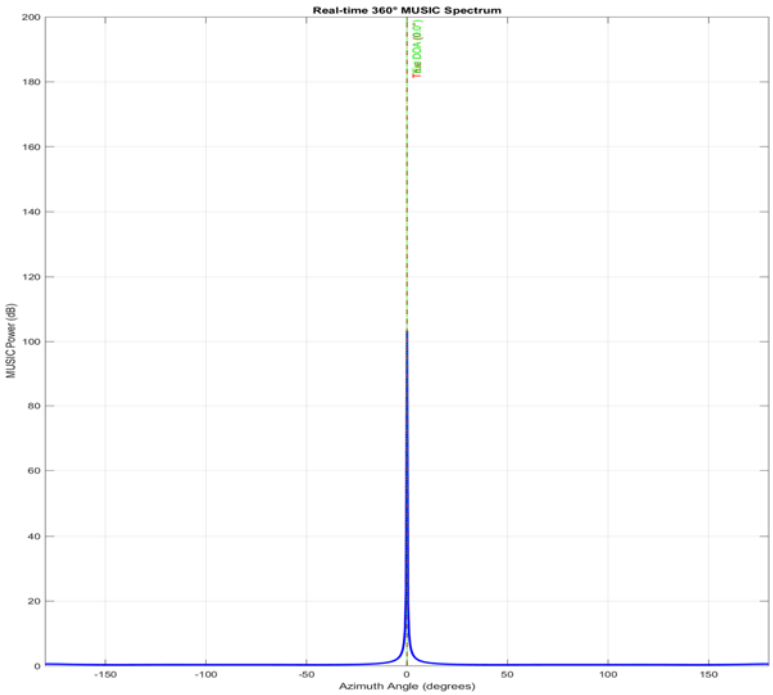


Figure 2. – Real-time 360° MUSIC spatial power spectrum indicating exact alignment between true target DOA

($\theta_{true} = 45.0°$) and estimated spatial peak ($\theta_{est} = 45.0°$)

The results confirm that combining direct Cartesian angle mapping (atan2d) with a 6-element UCA provides robust, un-aliased 360° source tracking without front-back directional ambiguities.

Bibliography

1. Schmidt R. O. Multiple emitter location and signal parameter estimation. *IEEE Transactions on Antennas and Propagation.* 1986. Vol. 34, No. 3. P. 276–280.

2. Ioannides P., Balanis C. A. Uniform circular arrays for smart antennas. *IEEE Antennas and Propagation Magazine.* 2005. Vol. 47, No. 4. P. 192–206.

3. Tuncer T. E., Friedlander B. (eds.). Classical and Modern Direction of Arrival Estimation. Academic Press, 2009. 456 p.

УДК 004.7:004.051

Ігор Москаленко
здобувач другого (магістерського) рівня вищої освіти
спеціальності «Комп'ютерна інженерія»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Олексій Тогоєв
PhD, старший викладач кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ПРОГРАМНО-АПАРАТНІ ЗАСОБИ АВТЕНТИФІКАЦІЇ В EDGE-ІНФРАСТРУКТУРІ СИСТЕМ БЕЗПЕКИ НА ОСНОВІ МІКРОКОМП'ЮТЕРА RASPBERRY PI

Стрімке зростання кількості автоматизованих точок контролю фізичного доступу на об'єктах критичної та корпоративної інфраструктури висуває якісно нові вимоги до архітектурних принципів побудови систем контролю та управління доступом (СКУД). Традиційна парадигма функціонування таких систем базується на централізованій обробці: периферійний зчитувач виконує роль простого перетворювача сигналу, передаючи зчитаний ідентифікатор через польові шинні інтерфейси (Wiegand, RS-485/OSDP) або локальну мережу до віддаленого центрального сервера чи хмарного сервісу. За такої організації центральний вузол приймає рішення щодо надання доступу та надсилає зворотну керуючу команду на виконавчий механізм.

Проте збільшення масштабу об'єктів та зростання інтенсивності потоків суб'єктів доступу виявляє фундаментальні недоліки централізованих архітектур. До них належать критична залежність від надійності каналів передавання даних, затримки мережевого відгуку, ризик утворення черг у пікові періоди та наявність єдиної точки відмови [5]. У разі втрати зв'язку із сервером точки проходу або блокуються, зупиняючи функціонування об'єкта, або переходять у незахищений аварійний режим.

Для подолання цих обмежень у галузі комп'ютерної інженерії активно впроваджується концепція периферійних (граничних) обчислень – Edge Computing [1; 6]. Суть парадигми полягає у перенесенні обчислювальних операцій, криптографічної перевірки

прав та прийняття рішень безпосередньо на периферійні мікропроцесорні вузли, розташовані в безпосередній близькості до виконавчих механізмів точок доступу. Це забезпечує мінімальні затримки спрацювання, автономність функціонування та суттєве розвантаження мережевої інфраструктури [5–6]. Водночас апаратні обмеження компактних Edge-контролерів вимагають розробки збалансованих програмно-апаратних засобів та протоколів ідентифікації, які поєднують високу криптографічну стійкість із помірним використанням системних ресурсів [1].

Ефективність та захищеність Edge-інфраструктури систем безпеки визначається використовуваним методом ідентифікації суб'єкта доступу. Серед найбільш поширених підходів вирізняють радіочастотну ідентифікацію (RFID), біометричні технології, оптичні коди та протоколи бездротового зв'язку малого радіуса дії [1; 3].

Технологія пасивного RFID на базі карт стандарту MIFARE Classic довгий час залишалася домінуючим рішенням завдяки низькій вартості та простоті інтеграції. Проте вразливість пропрієтарного алгоритму шифрування Crypto-1 і масове поширення портативних емуляторів призвели до критичного зниження захищеності: копіювання таких карт здійснюється зловмисниками безконтактно за лічені секунди. Перехід на стандарти MIFARE Plus або DESFire усуває математичні вразливості завдяки використанню алгоритмів AES-128, проте зберігає проблему фізичного відчуження носія – передачу карти сторонній особі або її втрату.

Біометричні методи (зокрема розпізнавання обличчя) виключають проблему відчуження ідентифікатора, однак для їх локальної реалізації на Edge-вузлах потрібні значні обчислювальні потужності [3]. Виконання згорткових нейромереж на компактних одноплатних комп'ютерах призводить до високої утилізації процесорних ядер, зростання тепловиділення, а за відсутності спеціалізованих апаратних NPU-прискорювачів – до затримок розпізнавання, що досягають 1-2 секунд [3; 6].

Перспективним напрямком є оптична автентифікація за допомогою динамічних матричних кодів (QR-кодів) [1–2]. Традиційна схема, за якої код генерується на смартфоні користувача і зчитується камерою контролера, вимагає безперервного захоплення відеопотоку та постійної роботи програмних алгоритмів детекції кадру, що створює значне фонове навантаження на процесор Edge-вузла.

Більш ефективною та криптографічно надійною є зворотна схема: сам Edge-вузол генерує на власному вбудованому TFT-дисплеї псевдовипадковий оптичний сесійний челендж із коротким життєвим

циклом [2]. Користувач сканує код камерою смартфона через спеціалізований застосунок, підписує його локально збереженим ключем і відправляє авторизаційний запит назад до Edge-вузла через локальний бездротовий канал. Це повністю знімає з периферійного мікрокомп'ютера обчислювальний тягар обробки відеопотоку, переносячи операції сканування на пристрій користувача, та гарантує захист від копіювання, перехоплення та атак повторного відтворення [1–2]. Порівняльну характеристику зазначених підходів за ключовими технічними критеріями наведено в табл. 1.

Таблиця 1. – Порівняльна характеристика підходів до автентифікації в Edge-інфраструктурі систем безпеки

Підхід	**Принцип автентифікації**	**Апаратне забезпечення Edge-вузла**	**Основна перевага**	**Основне обмеження**
Пасивний RFID (MIFARE Classic)	Зчитування статичного UID / блоку даних мітки	Зчитувальний модуль (RC522), інтерфейс SPI / UART	Мінімальні апаратні витрати та миттєвий відгук	Низька стійкість до клонування та емуляції
Криптографічний RFID (DESFire)	Взаємна автентифікація з шифруванням AES/3DES	NFC-контролер (PN532), інтерфейс I2C / SPI	Висока криптографічна стійкість каналу зчитування	Висока вартість носіїв, ризик передачі карти третім особам
Оптичне зчитування QR камерою	Сканування камерою коду з екрана смартфона	Модуль камери (CSI/USB), бібліотека детекції кадрів	Відсутність фізичних карт, зручність для користувача	Постійне високе завантаження CPU відеопотоком
Генерація QR на Edge-дисплеї	Відображення сесійного челенджу, сканування клієнтом	Графічний дисплей (TFT/DSI), локальний сервіс API	Розвантаження CPU, захист від атак повтору, інтерактивність	Необхідність використання клієнтського застосунку
Локальна біометрія (Face ID)	Захоплення обличчя та вилучення дескрипторів	HD-камера, нейромережевий співпроцесор (NPU)	Неможливість відчуження ідентифікатора	Висока вартість, енергоспоживання, ризик нагріву

Функціонування периферійного вузла системи безпеки супроводжується динамічним навантаженням на його апаратні підсистеми [6]. На відміну від серверних систем, обчислювальні

ресурси мікрокомп'ютера є обмеженими, тому процес автентифікації повинен оптимізуватися за кількома взаємопов'язаними факторами:

– періодичність генерації візуального челенджу: часовий інтервал Δt оновлення оптичного коду визначає рівень безпеки (чим менший інтервал, тим менше вікно для компрометації) та обчислювальні витрати на перемальовування кадрового буфера дисплея;

– інтенсивність потоку запитів: кількість сеансів автентифікації, що надходять до локального API Edge-вузла за одиницю часу у години пікового навантаження точки доступу;

– криптографічна трудомісткість: використання асиметричних алгоритмів цифрового підпису чи алгоритмів симетричного хешування для валідації челенджу;

– швидкодія виконавчого контуру: затримка між підтвердженням валідності токена та формуванням апаратного сигналу комутації на лініях введення-виведення загального призначення для розблокування замка.

– Для всебічного оцінювання ефективності системи безпеки на Edge-рівні доцільно виділити три взаємодоповнюючі групи показників [5–6]:

– ресурсні показники: коефіцієнт завантаження процесорних ядер, динаміка споживання оперативної пам'яті, навантаження на дискову підсистему при веденні локальних транзакційних журналів та температура кристала SoC;

– часові характеристики: затримка обробки авторизаційного запиту, час локальної верифікації криптографічного підпису та сумарний час відкриття проходу від моменту сканування;

– показники надійності та автономності: стабільність роботи у повністю ізольованому автономному режимі за відсутності зовнішньої мережі, коефіцієнт відмов при піковому опитуванні та гарантована відсутність витоків пам'яті за тривалої експлуатації.

Таким чином, ефективність Edge-вузла системи безпеки є комплексною характеристикою, що визначається здатністю забезпечувати надійний криптографічний контроль за мінімального часу відгуку та низьких витрат апаратних ресурсів.

Структура досліджуваного програмно-апаратного середовища. Для проведення практичних досліджень та апробації запропонованих підходів розроблено прототип автономного Edge-термінала безпеки на базі мікрокомп'ютера Raspberry Pi 4 Model B.

До складу апаратної частини вузла інтегровано графічний сенсорний дисплей, підключений за шиною DSI/SPI, який слугує як для відображення сесійного динамічного коду, так і для інтерактивного локального адміністрування системи (перегляд журналів подій, керування базою користувачів). Для керування виконавчим електромеханічним замком задіяно цифрові лінії GPIO, комутація яких здійснюється через блок гальванічної оптичної розв'язки, що запобігає впливу імпульсних наведень соленоїда на логічні кола мікрокомп'ютера [4].

Програмна складова побудована за модульним принципом у контейнеризованому середовищі під керуванням оптимізованої 64-розрядної операційної системи Linux. Структуру розробленого досліджуваного середовища представлено на рис. 1.



Рисунок 1. – Структура досліджуваного Edge-середовища автентифікації

Для оцінювання ресурсного стану Edge-вузла проаналізовано показники функціонування мікрокомп'ютера Raspberry Pi під час циклічного оновлення QR-коду на дисплеї ($\Delta t = 15$ с) та обробки потоку авторизаційних сесій (1-5 запитів/с). За даними моніторингу телеметрії, середнє завантаження процесора становило 4,25% (час простою – 95,75%), короткочасно зростаючи до 12,80% у моменти

пікової верифікації підписів. Використання оперативної пам'яті зафіксовано на стабільному рівні 116,4 Мбайт (близько 2,9% пулу), а очікування дискових операцій (I/O wait) не перевищувало 0,18%. Повний час від отримання запиту від смартфона до спрацювання реле замка GPIO склав $26{,}8 \pm 3{,}5$ мс при робочій температурі кристала 46-49°C [4; 6].

Отримані значення підтверджують високу обчислювальну стійкість комплексу та демонструють, що перенесення сканування на пристрій користувача вивільняє ресурси мікрокомп'ютера для криптографічного захисту й автономної роботи.

Список використаних джерел

1. Jevin J. T., Cini J., Aryadevi C., James M. Edge Computing. *International journal of engineering technology and management sciences*. 2022. Vol. 6. P. 567–573. DOI: 10.46647/ijetms.2022.v06i05.091
2. Zhang L., Luo T., Zhao H., Xiao C., Wang J. Software-Defined Runtime Reconfiguration of Cryptographic Service Chains for IIoT Edge Nodes. *Electronics*. 2026. Vol. 15, Issue 15, 3325. DOI: 10.3390/electronics15153325
3. Ali F., Mali S., Mahakud R., Yadav G.. AI-enabled smart surveillance system for secure monitoring and authentication. *Sci Rep*. 2026. Vol. 16. 21686. DOI:10.1038/s41598-026-52387-w
4. Raspberry Pi 4 Specifications and Hardware Deployment Documentation. Raspberry Pi Foundation. 2024. URL: https://www.raspberrypi.com/documentation/computers/raspberry-pi.html (дата звернення: 12.10.2026).
5. Alwanto H., Yel M. B. Analysis of enterprise network performance using the SNMP (Simple Network Management Protocol) method. *Journal Innovations Computer Science*. 2025. Vol. 4, Issue 2. P. 335–344. DOI: 10.56347/jics.v4i2.346
6. Cao J., Yu Z., Zhu B. Construction and efficiency analysis of an embedded system-based verification platform for edge computing. *Sci Rep.* 2025. Vol. 15. 26114. DOI:10.1038/s41598-025-10580-3

УДК 004.75

Данило Невідомий
здобувач другого (магістерського) рівня вищої освіти
спеціальності «Комп'ютерна інженерія»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Сергій Пузирьов
кандидат фіз.-мат. наук, доцент,
доцент кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

РОЗПОДІЛЕНА ХМАРНА СИСТЕМА ПАНОРАМНОГО ВІДЕОСПОСТЕРЕЖЕННЯ ЗА АРХІТЕКТУРОЮ EDGE-TO-CLOUD

Актуальність теми. У межах бакалаврської роботи Д. Невідомого було реалізовано програмно-апаратний комплекс об'єднання відеопотоків у реальному часі на базі мікрокомп'ютера Raspberry Pi, що ефективно вирішує задачу формування панорамного зображення на одному пристрої. Проте окремий автономний пристрій не дозволяє централізовано керувати мережею панорамних точок спостереження, збирати й зберігати дані з багатьох джерел та масштабувати рішення на рівні цілого об'єкта. Зростання попиту на системи відеоспостереження з широким кутом огляду зумовлює потребу в надійній серверній інфраструктурі, здатній агрегувати відеопотоки й телеметрію з багатьох пристроїв одночасно та комплексно ліквідовувати «сліпі зони».

Мета даної роботи – масштабувати створений прототип до рівня розподіленої хмарної системи панорамного відеоспостереження за архітектурою Edge-to-Cloud шляхом побудови надійної бекенд-інфраструктури, яка забезпечує агрегацію даних з кількох периферійних пристроїв, їх збереження та безпечний доступ користувачів, перетворюючи локальне рішення на повноцінний комерційно орієнтований програмний продукт.

Архітектура системи. В основу покладено дворівневу архітектуру Edge-to-Cloud (рис. 1). Нижній рівень утворює периферійний вузол (Edge-node) – апаратний стенд на базі Raspberry Pi 5, який виконує обчислювально складну обробку кадрів безпосередньо на пристрої,

зокрема розрахунок матриці гомографії та фотометричне згладжування швів. Завдяки цьому на верхній, серверний рівень передається вже зшитий відеопотік разом із системною телеметрією, що зменшує навантаження на мережу та централізований сервер.

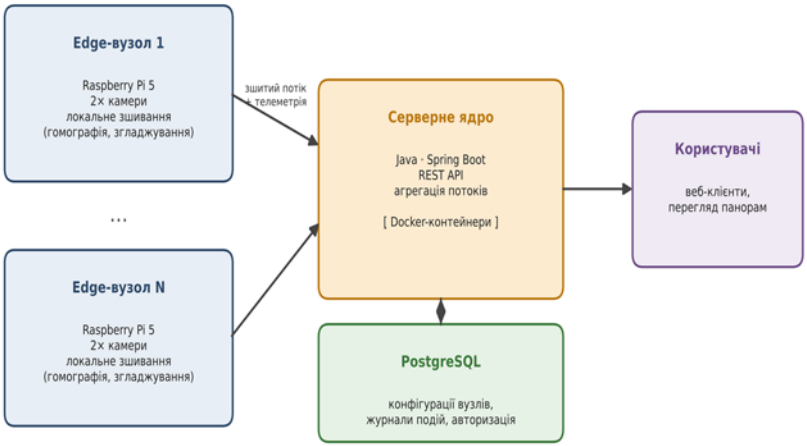


Рисунок 1. – Архітектура системи Edge-to-Cloud

Серверне ядро та програмна платформа. Серверне ядро системи розробляється мовою Java з використанням фреймворку Spring Boot, що дозволяє організувати стабільний REST API для одночасної агрегації даних з кількох панорамних пристроїв. Для збереження конфігурацій вузлів, логування подій та забезпечення безпечної авторизації користувачів інтегрується реляційна база даних PostgreSQL. Уся серверна частина пакується у Docker-контейнери, що забезпечує зручне та незалежне розгортання системи.

Принцип роботи. Кожен периферійний вузол безперервно передає зшитий відеопотік і телеметрію на централізований сервер через REST API. Серверне ядро приймає та агрегує дані від множини пристроїв, зберігає їхні конфігурації й журнали подій у PostgreSQL та контролює доступ користувачів через механізми авторизації. Контейнеризоване середовище дозволяє горизонтально масштабувати систему, під'єднуючи нові Edge-вузли без зміни архітектури сервера.

Практичне застосування. Розроблювана система орієнтована на: комплексне відеоспостереження за територіально розподіленими об'єктами; охорону периметрів та критичної інфраструктури;

транспортні вузли й системи «розумного міста»; промисловий моніторинг технологічних процесів.

Переваги розробки. Основними перевагами запропонованого рішення є: масштабованість за рахунок під'єднання довільної кількості Edge-вузлів; централізоване керування мережею пристроїв; розвантаження сервера завдяки винесенню важких обчислень на периферію; незалежне та відтворюване розгортання у Docker-контейнерах; безпечний доступ користувачів на основі авторизації.

Висновки. Запропонований архітектурний перехід від локального прототипу до розподіленої хмарної системи Edge-to-Cloud перетворює автономний пристрій на повноцінний комерційно орієнтований продукт для ліквідації «сліпих зон». Інтеграція периферійних систем комп'ютерного зору із сучасними серверними рішеннями на базі Java Spring Boot, PostgreSQL та Docker забезпечує надійність, масштабованість і централізоване керування панорамним відеоспостереженням.

Список використаних джерел

1. Walls C. Spring in Action. 6th ed. Shelter Island : Manning Publications, 2022. 520 p.
2. Spring Boot Reference Documentation. Spring. URL: https://docs.spring.io/spring-boot/ (дата звернення: 18.10.2026).
3. PostgreSQL 16 Documentation. The PostgreSQL Global Development Group. URL: https://www.postgresql.org/docs/ (дата звернення: 18.10.2026).
4. Docker Documentation. Docker Inc. URL: https://docs.docker.com/ (дата звернення: 18.10.2026).
5. Shi W., Cao J., Zhang Q., Li Y., Xu L. Edge Computing: Vision and Challenges. *IEEE Internet of Things Journal.* 2016. Vol. 3, No. 5. P. 637–646.
6. OpenCV: High level stitching API (Stitcher class). OpenCV documentation index. URL: https://docs.opencv.org/4.x/d8/d19/tutorial_stitcher.html (дата звернення: 18.10.2026).

УДК 004.7:004.051

Віталій Нікольський
доктор технічних наук, професор,
професор технічної експлуатації флоту
Національний університет «Одеська морська академія»
м. Одеса, Україна
Олександр Юрченко
здобувач другого (магістерського) рівня вищої освіти
спеціальності «Комп'ютерна інженерія»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ВИКОРИСТАННЯ ТЕЛЕМЕТРІЇ ДЛЯ ОЦІНЮВАННЯ ЕФЕКТИВНОСТІ СИСТЕМ МОНІТОРИНГУ МЕРЕЖЕВОЇ ІНФРАСТРУКТУРИ

Сучасні мережеві інфраструктури характеризуються постійним збільшенням кількості мережевих пристроїв, серверних систем, програмних компонентів та інших джерел інформації про стан інформаційної системи. Функціонування таких компонентів супроводжується формуванням значних обсягів телеметричних даних, які містять інформацію про продуктивність обладнання, стан мережевих інтерфейсів, використання обчислювальних ресурсів, доступність сервісів та інші параметри функціонування інфраструктури. Зі зростанням масштабу мережі та інтенсивності надходження даних актуальним стає не лише забезпечення їх збору, а й оцінювання ефективності відповідних процесів.

Телеметричні дані можуть надходити до системи моніторингу різними способами залежно від типу джерела, характеру інформації та вимог до оперативності її отримання. До найбільш поширених підходів належать періодичне опитування пристроїв, агентський збір показників, передача подій і журналів, отримання інформації через програмні інтерфейси, а також збір мережевої потокової інформації. Кожен із зазначених підходів має власні особливості, які необхідно враховувати під час побудови системи моніторингу та визначення її продуктивності [1].

Одним із поширених способів отримання телеметрії від мережевого обладнання є протокол SNMP, який дає змогу отримувати значення лічильників, параметрів стану та інших характеристик пристроїв [2−3]. Перевагою такого підходу є його широке застосування у мережевому

обладнанні різних виробників. Водночас збільшення кількості опитуваних параметрів та скорочення інтервалу між запитами призводить до зростання кількості операцій збору та, відповідно, навантаження на систему моніторингу.

Агентський підхід передбачає використання програмного компонента, встановленого на контрольованому сервері або робочій станції. Агент може отримувати локальні показники операційної системи та прикладних компонентів і передавати їх до системи моніторингу. Такий підхід забезпечує доступ до ширшого набору параметрів, які складно або неможливо отримати за допомогою універсальних мережевих протоколів. Водночас масштабування кількості контрольованих систем збільшує кількість джерел телеметрії та обсяг даних, які повинна обробляти центральна система. Окремий клас становлять подієві та потокові механізми збору. На відміну від періодичного опитування, вони можуть формувати інформаційний потік у відповідь на виникнення певної події або відповідно до поточної активності мережі. До таких підходів належать передавання системних і прикладних журналів, а також технології експорту інформації про мережеві потоки. Їх використання дає змогу отримувати додаткову інформацію про функціонування інфраструктури, проте за високої інтенсивності подій може формувати значне навантаження на канали передачі, компоненти обробки та системи зберігання [4].

Таким чином, ефективність збору телеметричних даних не може визначатися лише можливістю отримати необхідну інформацію. Доцільно враховувати сукупність кількісних і якісних характеристик процесу збору. До основних критеріїв можна віднести продуктивність обробки даних, використання обчислювальних ресурсів, мережеве навантаження, вимоги до дискової підсистеми, затримку отримання інформації та масштабованість системи. Для систематизації основних підходів до збору телеметричних даних доцільно порівняти їх за характером отримання інформації, типом даних, вимогами до інфраструктури збору та особливостями масштабування. Результати такого порівняння наведено в табл. 1.

Таблиця 1. – Порівняльна характеристика підходів до збору телеметричних даних

Підхід	Принцип збору	Тип телеметричних даних	Основна перевага	Основне обмеження
Періодичне опитування	Регулярне надсилання запитів до джерела	Показники стану, лічильники, параметри продуктивності	Простота організації та прогнозованість потоку даних	Навантаження зростає зі збільшенням частоти опитування
Агентський збір	Отримання даних програмним агентом на контрольованому вузлі	Параметри ОС, апаратних ресурсів і застосунків	Висока деталізація локальних показників	Необхідність встановлення та супроводу агента
Подієвий збір	Передача інформації після виникнення події	Журнали, повідомлення про події та помилки	Оперативне отримання інформації про події	Нерівномірний та потенційно значний потік даних
Потоковий збір	Передача інформації про мережеву активність до колектора	Дані про мережеві потоки та характеристики трафіку	Можливість аналізу структури та інтенсивності мережевого трафіку	Значні вимоги до обробки та зберігання при високій інтенсивності
API-інтеграція	Отримання даних через програмний інтерфейс	Структуровані показники та дані інформаційних систем	Гнучкість інтеграції з різними системами	Залежність від можливостей і продуктивності конкретного API

Наведені підходи відрізняються не лише способом отримання інформації, а й характером навантаження, яке вони формують на компоненти системи моніторингу. Для періодичного та агентського збору визначальними параметрами є кількість контрольованих показників і частота їх оновлення, тоді як для подієвих та потокових механізмів значний вплив має інтенсивність виникнення подій або мережевої активності. Тому оцінювання ефективності збору телеметрії потребує врахування як характеристик джерел даних, так і ресурсів, необхідних для їх передачі, обробки та зберігання.

Одним із ключових факторів, що визначає навантаження на систему збору телеметричних даних, є інтенсивність надходження інформації. Вона залежить від кількості джерел телеметрії, кількості

показників, що контролюються для кожного джерела, та періодичності їх отримання. У разі періодичного збору збільшення кількості контрольованих показників або зменшення інтервалу між запитами призводить до збільшення кількості операцій, які повинна виконувати система моніторингу. Для великих інфраструктур це може суттєво впливати на використання процесорних ресурсів, оперативної пам'яті, мережевої пропускної здатності та дискової підсистеми.

Окреме значення має показник кількості нових значень, що надходять до системи за одиницю часу. Він дає змогу пов'язати конфігураційні параметри моніторингу з фактичним обсягом телеметричного потоку та використовуватися для порівняння різних режимів роботи системи. Зі збільшенням цього показника очікується відповідне зростання навантаження на компоненти збору та обробки, а також на підсистему зберігання даних. При цьому характер такої залежності може відрізнятися залежно від використовуваного методу збору, типу телеметрії та архітектури системи.

Для комплексного оцінювання ефективності доцільно враховувати декілька груп показників. До ресурсних показників належать завантаження центрального процесора, використання оперативної пам'яті та дискової підсистеми. Мережеві характеристики можуть оцінюватися за обсягом переданих даних, кількістю мережевих операцій та затримкою отримання інформації. Характеристики самого процесу моніторингу можуть включати кількість нових значень за одиницю часу, довжину черг необроблених даних та своєчасність отримання телеметрії [5].

Таким чином, ефективність системи збору доцільно розглядати як комплексну характеристику, яка визначається не одним показником, а співвідношенням обсягу та актуальності отримуваної телеметрії до ресурсів, необхідних для її збору, обробки та зберігання. Такий підхід дозволяє перейти від простого визначення доступності телеметричних даних до кількісного оцінювання ефективності функціонування системи моніторингу.

Для практичного дослідження використано систему моніторингу Zabbix 7.4.15, яка забезпечує збір, обробку та зберігання телеметричних даних [2; 6]. Структуру досліджуваного середовища наведено на рис. 1.

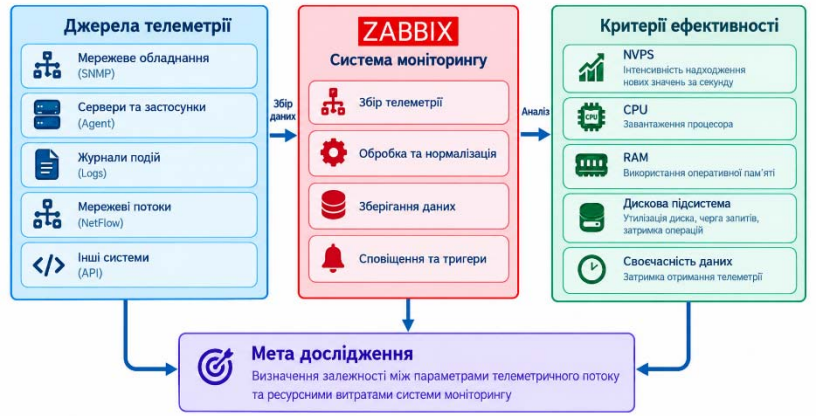


Рисунок 1. – Структура досліджуваного середовища та критерії оцінювання ефективності збору телеметричних даних

Для оцінювання ресурсного стану досліджуваного середовища проаналізовано показники завантаження центрального процесора (CPU), використання оперативної пам’яті (RAM), стану дискової підсистеми та інтенсивності надходження телеметричних даних (NVPS, New Values Per Second) – кількість нових значень за секунду. За даними моніторингу, завантаження центрального процесора становило 3,18%, а частка часу простою – 96,82%. Показник очікування операцій введення-виведення дорівнював 0,29%.

Для дискового пристрою /dev/sda (SCSI Disk A), позначення якого використовується операційною системою Linux, зафіксовано утилізацію на рівні 2,49 %, середній розмір черги запитів – 0,058, а середній час очікування операцій запису близько 0,40 мс. Використання кореневої файлової системи становило 28,32% від загального обсягу 63,44 Гбайт.

Отримані значення характеризують стан ресурсів у момент спостереження та свідчать про невисоке поточне навантаження на контрольовані компоненти. Водночас одного знімка показників недостатньо для визначення граничної продуктивності системи або встановлення залежності між інтенсивністю збору телеметрії та ресурсними витратами. Для обґрунтованого оцінювання необхідно дослідити динаміку показників упродовж визначеного інтервалу та зіставити її зі змінами інтенсивності надходження даних.

Список використаних джерел

1. RFC 9232: network telemetry framework. RFC Editor. URL: https://www.rfc-editor.org/rfc/rfc9232.html (дата звернення: 09.10.2026).
2. Zabbix documentation 7.4. URL: https://www.zabbix.com/ documentation/ 7.4/en/manual (дата звернення: 10.10.2026).
3. Alwanto H., Yel M. B. Analysis of enterprise network performance using the SNMP (Simple Network Management Protocol) method. *Journal Innovations Computer Science.* 2025. Vol. 4, Issue 2. P. 335–344. DOI: 10.56347/jics.v4i2.346
4. Documentation. OpenTelemetry. URL: https://opentelemetry.io/ docs/ (дата звернення: 10.10.2026).
5. Signals. OpenTelemetry. URL: https://opentelemetry.io/docs /concepts/ signals/ (дата звернення: 08.10.2026).
6. Онищук О. Порівняння інструментів системного моніторингу Nagios, Zabbix, Prometheus, MS SCOM для виявлення аномалій та проблем в мережі. *Кібербезпека: освіта, наука, техніка.* 2026. Том 1, № 33. С. 313–319. DOI: 10.28925/2663-4023.2026.33.1132

УДК 004.4:681.5

Андрій Павлюк
здобувач другого (магістерського) рівня вищої освіти
спеціальності «Комп'ютерна інженерія»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євген Дарнапук
PhD, доцент,
в.о. завідувача кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ПРОГРАМНА АРХІТЕКТУРА ТА АЛГОРИТМ ОЦІНЮВАННЯ EDGE-AI КЕРУВАННЯ МОБІЛЬНОЮ ПЛАТФОРМОЮ

Актуальність теми. Програмне забезпечення автономної мобільної платформи має узгоджувати обробку зображень із формуванням команд приводам. Короткий час нейромережевого

виведення не характеризує очікування в черзі кадрів, підготовку даних і передавання результату контролеру. На вбудованому комп'ютері ці операції конкурують із системними процесами. Дослідження DeepPicar підтверджує вплив конкуренції за спільні ресурси на час виконання нейромережевого керування [1]. Тому програмні конфігурації потрібно оцінювати за поведінкою всього контуру.

Мета даної роботи – визначити модульну структуру програмного забезпечення Edge-AI керування та алгоритм оцінювання його часових характеристик на основі записів окремих циклів обробки кадрів.

Програмна архітектура. Пропонується розділити отримання кадрів, попередню обробку, нейромережеве виведення, обчислення команд і журналювання. Модулі обмінюються даними зі спільним ідентифікатором кадру. Це дозволяє простежити, яке спостереження використано для конкретної команди, та змінювати рушій виведення без зміни інтерфейсу контролера. Для прототипу на NVIDIA Jetson Nano можна використати Python і бібліотеки середовища JetPack. Зокрема, JetPack 4.6.1 підтримує цю платформу та містить TensorRT 8.2.1 і CUDA 10.2 [2]. Версії компонентів потрібно зазначати в описі експерименту.

На рисунку 1 наведено структуру програмного конвеєра. Робочі дані передаються між модулями керування, а часові події та ресурсні відліки надходять до окремого журналу. Пунктир позначає збирання даних для аналізу.

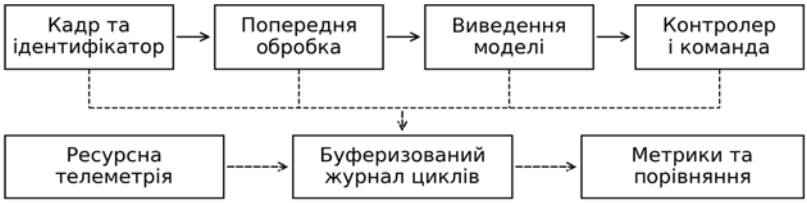


Рисунок 1 – Програмний конвеєр керування та збирання даних

Алгоритм циклу керування. На початку циклу модуль введення отримує кадр, присвоює йому ідентифікатор і фіксує час доступності застосунку. Далі виконуються зміна розміру, перетворення каналів і нормалізація відповідно до вимог моделі. Параметри обробки мають збігатися з використаними під час її навчання. Результат виведення передається контролеру, який обчислює команду з урахуванням

заданих обмежень та надсилає її на вихідний інтерфейс. Запис циклу завершується часовою міткою передавання команди.

Для зменшення накопичення застарілих спостережень доцільно обмежувати місткість черги та явно задавати політику вибору кадру. Наприклад, обробка найновішого доступного кадру може скоротити його очікування, але збільшує кількість пропусків. Ідентифікатори та час очікування потрібно зберігати для того, щоб виявляти цей компроміс. Контролер також має визначену реакцію на відсутню або прострочену команду; її перевіряють окремими тестами.

Вимірювання часу. Події реєструються на спільній монотонній часовій шкалі. Для реалізації на Python придатна time.monotonic(), яка не залежить від коригування системного календарного часу [3]. Основні інтервали визначають за формулою (1).

$$T_k = t_k^{out} - t_k^{in}, \; A_k = t_k^{out} - t_{k-1}^{out} \quad (1)$$

де T_k – час від доступності кадру до передавання відповідної команди; A_k – інтервал між сусідніми командами.

Експозиція до початкової мітки та фізична реакція приводу після кінцевої до T_k не входять. Для асинхронного конвеєра тривалості окремих етапів можуть перекриватися, тому їх сума не замінює наскрізного вимірювання.

GPU-операції можуть виконуватися асинхронно. Вимірювання лише часу повернення виклику може відображати постановку роботи в чергу, а не завершення обчислення. Для профілювання використовують CUDA-події або обґрунтовану синхронізацію [4]. GPU-тривалості зберігають окремо від часових міток застосунку; їх не можна безпосередньо змішувати як відліки одного годинника.

Записи циклів і показники. До запису циклу включають ідентифікатор кадру, часові мітки, результат виведення, вихідну команду та ознаку перевищення дедлайну. Ресурсна телеметрія має власні мітки й частоту опитування. Її пов’язують із циклом за заданим правилом і допустимою часовою відстанню. Низька частота моніторингу не дозволяє відновити короткі піки навантаження. Пропущені значення позначають як відсутні, а не як нулі.

Для контролера важливі не лише загальна кількість прострочень, а й їх послідовності: зв’язок послідовних пропусків дедлайнів зі стійкістю системи досліджено в [5]. У таблиці 1 наведено показники для запропонованої процедури.

Таблиця 1. – Показники програмного оцінювання

Показник	**Дані для розрахунку**	**Що характеризує**
Час циклу T_k	Вхідна й вихідна мітки	Середню затримку, 95-й процентиль і стандартне відхилення
Інтервал A_k	Мітки сусідніх команд	Регулярність оновлення керування
Пропуски дедлайну	T_k та допустимий час	Частку перевищень і найдовшу їх послідовність
Черга кадрів	Ідентифікатори й мітки	Очікування кадрів і кількість відкинутих спостережень
Вплив журналювання	Запуски з моніторингом і без нього	Додаткові витрати реєстрації даних

Порядок порівняння конфігурацій. Порівнюють конфігурації моделі, рушія виведення й розміру входу за незмінних контролера та сценарію руху. Спочатку перевіряють коректність передавання даних і реакції на пропуски, потім – часові характеристики. Після оптимізації контролюють похибку виходів моделі на відкладених даних. Для кожного запуску зберігають конфігурацію, виконують прогрівання та повторні вимірювання. Метрики розраховують окремо за запусками; послідовності пропусків не об’єднують через їхні межі.

Прийнятність визначають за заданими часовими й ресурсними обмеженнями. Скорочення затримки не має приховувати погіршення точності моделі або втрату команд. За відсутності незалежного вимірювання траєкторії висновки обмежують програмною своєчасністю. Журналювання виконують буферизовано та окремо оцінюють його накладні витрати.

Висновки. Визначено модульну архітектуру Edge-AI керування та послідовність оцінювання її програмних конфігурацій. Ідентифікатори кадрів і монотонні часові мітки дозволяють простежити обробку спостереження до відповідної команди. Профілювання етапів, контроль черги та аналіз пропусків дедлайнів дають змогу перевіряти наслідки програмної оптимізації на рівні всього контуру. Запропоновану процедуру потрібно реалізувати й перевірити на фізичній платформі; експериментальні показники швидкодії тут не наводяться.

Список використаних джерел

1. Bechtel M. G., McEllhiney E., Kim M., Yun H. DeepPicar: A Low-Cost Deep Neural Network-Based Autonomous Car. *2018 IEEE 24th International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA).* 2018. P. 11–21. DOI: 10.1109/RTCSA.2018.00011
2. JetPack SDK 4.6.1. NVIDIA Developer. URL: https://developer.nvidia.com/embedded/jetpack-sdk-461 (дата звернення: 19.10.2026).
3. time – Time access and conversions. Python documentation. URL: https://docs.python.org/3/library/time.html (дата звернення: 19.10.2026).
4. CUDA C++ Best Practices Guide. CUDA Toolkit 10.2.89. NVIDIA. 2019. URL: https://docs.nvidia.com/cuda/archive/10.2/cuda-c-best-practices-guide/index.html (дата звернення: 19.10.2026).
5. Maggio M., Hamann A., Mayer-John E., Ziegenbein D. Control-System Stability Under Consecutive Deadline Misses Constraints. *32nd Euromicro Conference on Real-Time Systems (ECRTS 2020).* 2020. Vol. 165. P. 21:1–21:24. DOI: 10.4230/LIPIcs.ECRTS.2020.21

УДК 004.056

Поліна Пахомова
здобувач першого (бакалаврського) рівня вищої освіти
спеціальності «Комп'ютерна інженерія»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євген Дарнапук
PhD, доцент,
в.о. завідувача кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ПЛАТФОРМА АВТОМАТИЗОВАНОГО ФОРМУВАННЯ СПЕЦИФІКАЦІЙ ГІБРИДНИХ МЕРЕЖЕВИХ ПРОТОКОЛІВ ІЗ ПОСТКВАНТОВОЮ КРИПТОГРАФІЄЮ

Актуальність теми. Розвиток квантових обчислень зумовлює потребу в перегляді механізмів захисту мережевих з'єднань. Достатньо потужний квантовий комп'ютер становитиме загрозу для RSA та криптографічних схем на еліптичних кривих.

Одним із підходів до переходу на постквантову криптографію є гібридні схеми, які поєднують традиційні та постквантові алгоритми [1]. Їхнє упровадження потребує узгодження форматів повідомлень, правил обробки та параметрів безпеки.

Збільшення розмірів ключів, шифротекстів або підписів може збільшити обсяг службового трафіку та час встановлення з'єднання. Вплив на продуктивність залежить від алгоритмів, реалізації та умов мережі. Тому доцільно автоматизувати формування специфікацій і порівняння конфігурацій до їх упровадження, а не припускати, що накладні витрати відсутні.

Мета даної роботи – розробити концепцію платформи Post-Quantum Network Protocol Generator для автоматизованого формування специфікацій гібридних мережевих протоколів, генерації тестових реалізацій та оцінювання їхніх накладних витрат у контрольованому середовищі.

Апаратна та програмна платформа. Платформу передбачається реалізувати на робочій станції або сервері під керуванням Linux. Програмна частина має поєднувати модулі опису вимог, генерації специфікацій і запуску експериментів із бібліотеками криптографічних алгоритмів. Як основу для постквантових операцій доцільно розглядати стандартизовані ML-KEM для встановлення спільного секрету та ML-DSA для цифрового підпису [2–3]. Ці механізми виконують різні функції та не є взаємозамінними. Загальну послідовність роботи наведено на рис. 1.

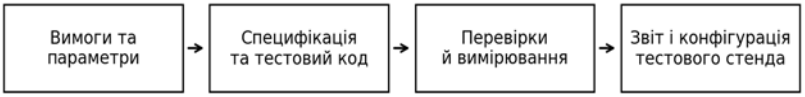


Рисунок 1. – Послідовність роботи запропонованої платформи

Принцип роботи. Платформа має виконувати такі етапи:

1. Введення вимог. Користувач задає призначення протоколу, допустимі алгоритми, обмеження ресурсів і параметри мережі. Окремо визначаються вимоги до конфіденційності та автентифікації.

2. Формування специфікації. На основі перевірених шаблонів генеруються формати повідомлень, порядок обміну та правила обробки помилок. Синтаксис описує структуру повідомлень, а семантика – їх значення та допустимі переходи між станами. Для гібридного встановлення ключів потрібен визначений механізм

поєднання секретів; довільне комбінування алгоритмів не гарантує безпеки.

3. Перевірка та оцінювання. Передбачаються перевірки відповідності повідомлень специфікації, сумісності вузлів і реакції на некоректні дані. У тестовій мережі вимірюються час установлення з'єднання, обсяг службового трафіку, корисна пропускна здатність і використання ресурсів. Результати слід порівнювати з традиційною конфігурацією за однакових умов і повторних запусків.

4. Формування вихідних матеріалів. Платформа має створювати специфікацію, тестовий код, конфігурацію стенда та звіт про вимірювання. Функціональні тести й моделювання продуктивності не замінюють криптографічного аналізу. Використання поза тестовим середовищем потребує окремої перевірки безпеки та сумісності.

Практичне застосування. Запропонована платформа орієнтована на навчальні й дослідницькі стенди, порівняння гібридних конфігурацій та підготовку до модернізації корпоративних мереж. Її використання може допомогти оцінити ресурсні вимоги для переходу на постквантові механізми.

Переваги розробки. Очікуваними перевагами є повторюваність експериментів, уніфікація специфікацій і зменшення обсягу ручного налаштування. Автоматизація може знизити ризик типових помилок конфігурації, проте не усуває його повністю. Зниження накладних витрат має бути підтверджене вимірюваннями.

Висновки. Запропоновано концепцію платформи для формування специфікацій і тестування гібридних мережевих протоколів із постквантовими механізмами. Визначено основні етапи обробки та показники оцінювання. Наступними завданнями є реалізація прототипу, перевірка коректності обміну та проведення порівняльних експериментів. Експериментальних результатів у цих тезах не наведено, тому висновки щодо продуктивності та захищеності реалізації потребують подальшого дослідження.

Список використаних джерел

1. Driscoll F., Parsons M., Hale B. Terminology for Post-Quantum Traditional Hybrid Schemes. RFC 9794. 2025. DOI: 10.17487/RFC9794

2. National Institute of Standards and Technology. Module-Lattice-Based Key-Encapsulation Mechanism Standard. FIPS 203. 2024. DOI: 10.6028/NIST.FIPS.203

3. National Institute of Standards and Technology. Module-Lattice-Based Digital Signature Standard. FIPS 204. 2024. DOI: 10.6028/NIST.FIPS.204

УДК 004.932:681.5

Яніс Смолянік
аспірант спеціальності
«Автоматизація, комп'ютерно-інтегровані технології
та робототехніка»
Сергій Пузирьов
кандидат фіз.-мат. наук, доцент,
доцент кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

АВТОМАТИЗОВАНЕ ВИЗНАЧЕННЯ ВІДСТАНІ ДО ПЕРЕШКОД ЗА МОНОКУЛЯРНИМ ЗОБРАЖЕННЯМ У СИСТЕМАХ КЕРУВАННЯ РУХОМИМИ ОБ'ЄКТАМИ НА БАЗІ RASPBERRY PI

Автономні мобільні платформи, складські візки, сільськогосподарські роботи та навчальні макети транспортних засобів потребують безперервної інформації про простір попереду за напрямом руху. Для системи автоматичного керування самого факту виявлення перешкоди замало: регулятор швидкості й модуль планування траєкторії оперують числовим значенням відстані, від якого залежить момент початку гальмування або об'їзду. Ця задача зазвичай може розв'язуватися за допомогою лідарів, ультразвукових далекомірів чи стереокамер, проте для малогабаритних платформ із жорсткими обмеженнями за вартістю, масою та енергоспоживанням привабливішим виглядає варіант з однією камерою, яка водночас використовується і для розпізнавання об'єктів. Порівняльні дослідження систем технічного зору [1] свідчать, що монокулярні схеми поступаються стереоскопічним за точністю, однак за коректного калібрування забезпечують результат, прийнятний для багатьох прикладних задач.

Метою даної роботи є розроблення та експериментальна перевірка методу автоматизованого визначення відстані до перешкоди за зображенням з однієї камери, реалізованого на одноплатному комп'ютері Raspberry Pi з бібліотекою OpenCV і придатного для включення в контур керування рухомим об'єктом.

Монокулярне вимірювання відстані зазвичай будують одним із двох способів. Перший спирається на подібність трикутників у моделі камери-обскури: якщо реальна висота об'єкта відома, відстань

знаходять через відношення добутку фокусної відстані на цю висоту до висоти зображення об'єкта в пікселях [2]. Для перешкод довільної форми й розміру такий підхід малопридатний, адже апріорних даних про об'єкт система не має. Другий спосіб, узятий нами за основу, виходить з того, що рухомий об'єкт і перешкода розташовані на одній площині, а камера закріплена на відомій висоті. Тоді відстань визначається положенням нижньої межі перешкоди на кадрі, тобто лінії її контакту з підлогою чи дорожнім покриттям. Подібну геометрію застосовано в системі адаптивного круїз-контролю з однією камерою [3], де автори також оцінили граничну точність такої схеми.

Схему вимірювання наведено на рис. 1. Камера встановлена на висоті h над площиною руху, а її оптична вісь нахилена до горизонту на кут θ. Якщо точка контакту P проєктується в рядок зображення y_b, а головна точка кадру має координату c_y, то промінь OP утворює з оптичною віссю кут $\alpha = \operatorname{arctg}((y_b - c_y)/f_y)$, і відстань до перешкоди становить

$$D = \frac{h}{\left(\theta + \frac{y_b - c_y}{f_y}\right)}, \tag{1}$$

де f_y – фокусна відстань камери в пікселях за вертикальною віссю. У нашій установці оптична вісь паралельна підлозі ($\theta = 0$), тому вираз (1) спрощується до вигляду

$$D = \frac{f_y h}{y_b - c_y}. \tag{2}$$

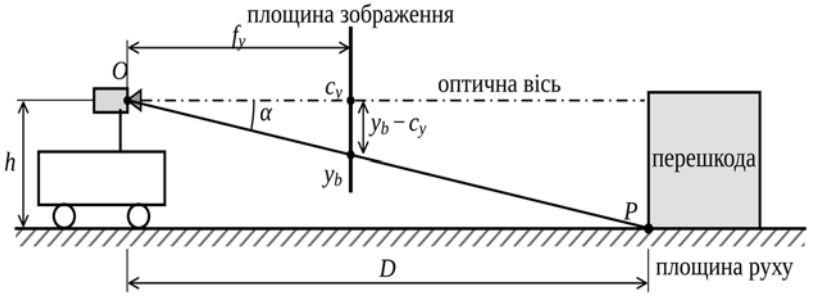


Рисунок 1. – Геометрична схема визначення відстані за точкою контакту перешкоди з площиною руху

Параметри f_y і c_y, а також коефіцієнти радіальної дисторсії об'єктива визначено калібруванням за методом Чжана [4], реалізованим у функції cv2.calibrateCamera. Для цього використано шахову мішень із 9×6 внутрішніми кутами та кроком клітинки 25 мм, знятої під двадцятьма різними ракурсами. Для камери Raspberry Pi Camera Module v2 у режимі 640×480 пікселів отримано f_y = 526,8 пкс і c_y = 241,3 пкс за середньоквадратичної похибки перепроєціювання 0,31 пкс. Корекцію дисторсії ми застосовуємо не до всього кадру, а лише до координат знайденої точки контакту через cv2.undistortPoints, що помітно зменшує навантаження на процесор.

Пошук перешкоди виконується в нижній половині кадру, яка відповідає зоні безпосередньо перед платформою. Зображення переводиться у відтінки сірого й згладжується фільтром Гаусса, після чого детектор Канні формує карту меж, а морфологічне замикання з'єднує розірвані фрагменти контурів. Контури з малою площею відкидаються як шум, а для решти визначається найнижча точка обмежувального прямокутника, яка й підставляється у формулу (2). Для того, щоб зменшити розкид значень між сусідніми кадрами, результат згладжується ковзним середнім за п'ятьма кадрами. На Raspberry Pi 4 Model B (4 ГБ) з OpenCV 4.8 повний цикл обробки одного кадру триває близько 47 мс, що відповідає приблизно 21 кадру за секунду.

Диференціювання виразу (2) дає оцінку похибки відстані, яка виникає через помилку визначення рядка y_b на Δy пікселів:

$$\Delta D \approx \frac{D^2 \Delta y}{f_y h}. \qquad (3)$$

З (3) видно, що похибка зростає пропорційно квадрату відстані, тоді як збільшення висоти встановлення камери чи її фокусної відстані цю похибку зменшує. За h = 0,15 м помилка в один піксель на відстані 1 м дає відхилення близько 1,3 см, а на відстані 3 м – уже 11 см. Та сама геометрія обмежує робочий діапазон і знизу: при горизонтальній оптичній осі камера не бачить підлогу ближче ніж $f_y h/(N_y - c_y) \approx 0{,}33$ м, де N_y = 480 – кількість рядків кадру.

Перевірку проводили в приміщенні з рівною підлогою, тестовою перешкодою слугував картонний короб заввишки 0,20 м. Для кожної контрольної відстані, виміряної рулеткою, фіксували 20 послідовних результатів, за якими обчислювали середнє значення $D_{\text{сер}}$, відносну похибку δ та середньоквадратичне відхилення σ. Паралельно для порівняння розраховувалася відстань за відомою висотою короба. Отримані дані наведено в табл. 1.

Таблиця 1 – Результати вимірювання відстані до тестової перешкоди

Фактична відстань *D*, м	**За відомою висотою об'єкта**		**За точкою контакту з площиною руху**		
	$D_{\text{сер}}$, м	δ, %	$D_{\text{сер}}$, м	δ, %	σ, м
0,50	0,507	1,4	0,506	1,2	0,004
1,00	0,988	1,2	1,018	1,8	0,009
1,50	1,523	1,5	1,537	2,5	0,019
2,00	2,041	2,1	2,064	3,2	0,034
2,50	2,446	2,2	2,598	3,9	0,056
3,00	3,084	2,8	3,157	5,2	0,083

Метод за відомою висотою на всьому діапазоні дав похибку не більше 2,8%, але працює він лише тоді, коли розмір об'єкта задано наперед. Метод за точкою контакту не потребує жодних відомостей про перешкоду; до 2 м його похибка не перевищує 3,2%, а на 3 м зростає до 5,2%, що добре узгоджується з квадратичною залежністю (3). Систематичне завищення результатів пояснюється тим, що тінь і розмиття біля основи короба зміщують виявлену нижню межу контуру на один-два пікселі вгору відносно справжньої лінії контакту. Цю складову можна значною мірою компенсувати поправкою, визначеною на етапі налаштування системи.

Для системи керування важлива не лише сама відстань, а й момент, коли потрібно подати команду на гальмування. Порогову відстань пропонуємо визначати з урахуванням швидкості руху v, затримки обробки τ, сповільнення a та похибки вимірювання:

$$D_{\text{пор}} = v\tau + \frac{v^2}{2a} + \Delta D, \tag{4}$$

де ΔD – похибка вимірювання на цій відстані.

Для платформи, що рухається зі швидкістю 0,5 м/с, при затримці 0,05 с і сповільненні 0,8 м/с^2 розрахунок дає близько 0,18 м. Це значення менше за мертву зону камери, тож в алгоритмі керування

поріг зупинки встановлено на рівні 0,5 м, аби рішення ухвалювалося, поки перешкода гарантовано перебуває в полі зору.

Проведене дослідження показало, що одна відкалібрована камера в поєднанні з Raspberry Pi здатна визначати відстань до перешкод довільної форми в діапазоні 0,5-3 м з відносною похибкою не більше 5,2% при частоті оновлення близько 21 кадру за секунду. Такої точності достатньо для формування керувальних команд у системах автоматичного запобігання зіткненням малогабаритних мобільних платформ. Подальшу роботу планується спрямувати на компенсацію коливань кута нахилу камери під час руху та на поєднання вимірювань з даними ультразвукового далекоміра.

Список використаних джерел

1. Барсов В., Костерна О., Плахотний О. Дослідження ефективності методів визначення відстані і геометричних параметрів об'єкта систем технічного зору. *Сучасні інформаційні системи*. 2020. Том 4, № 4. С. 64–69. DOI: 10.20998/2522-9052.2020.4.09

2. Hartley R., Zisserman A. *Multiple View Geometry in Computer Vision*. 2nd ed. Cambridge : Cambridge University Press, 2004. 655 p. DOI: 10.1017/CBO9780511811685

3. Stein G. P., Mano O., Shashua A. Vision-based ACC with a single camera: bounds on range and range rate accuracy. *IEEE IV2003 Intelligent Vehicles Symposium : Proceedings* (Columbus, OH, USA, 9–11 June 2003). 2003. P. 120–125. DOI: 10.1109/IVS.2003.1212895

4. Zhang Z. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2000. Vol. 22, No. 11. P. 1330–1334. DOI: 10.1109/34.888718

УДК 004.021

В'ячеслав Старченко
старший викладач кафедри комп'ютерної інженерії,
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ЕТОЛОГІЧНИЙ ПІДХІД ДО РОЗРОБКИ МОДЕЛЕЙ ПОВЕДІНКИ РОБОТІВ

Сучасна робототехніка все частіше стикається із завданнями, розв'язання яких вимагає не жорстко запрограмованої, а адаптивної, контекстно-залежної поведінки автономних систем. Роботи дедалі активніше функціонують у слабо структурованому, динамічному та частково непередбачуваному середовищі – від сервісних і соціальних роботів до мобільних платформ, що працюють поруч із людиною. За таких умов класичні підходи до планування поведінки, засновані на повній апріорній моделі світу та централізованому прийнятті рішень, демонструють обмежену масштабованість і недостатню робастність. Водночас біологічна еволюція вже виробила ефективні механізми адаптивної поведінки живих організмів у природному середовищі, детально описані в межах етології – науки про поведінку тварин [1]. Це створює природну методологічну передумову для перенесення етологічних принципів у інженерну практику розробки моделей поведінки роботів.

Основна проблема полягає у відсутності уніфікованого, теоретично обґрунтованого підходу до побудови архітектур поведінки автономних роботів, який би одночасно забезпечував:

– реактивність і швидкість реагування на зміни середовища;

– гнучкість та можливість пріоритезації конкуруючих поведінкових цілей;

– масштабованість системи без необхідності повного перепроєктування при додаванні нової функціональності;

– природність взаємодії з людиною в соціальних сценаріях застосування.

Наявні інженерні рішення (скінченні автомати, дерева поведінки, централізоване планування) вирішують ці задачі здебільшого фрагментарно й без єдиної концептуальної основи, тоді як етологія пропонує цілісний апарат опису мотивації, конкуренції поведінкових патернів та комунікативної взаємодії, що потребує систематичної адаптації до інженерного контексту.

Мета даної роботи – обгрунтувати доцільність та розглянути практичні механізми застосування етологічних концепцій (мотивація, ієрархія потреб, фіксовані комплекси дій, реліз-механізми, конкуренція поведінкових патернів) для побудови моделей поведінки автономних роботів.

Новизна роботи полягає у систематизації розрізнених етологічних концепцій у цілісну методологічну рамку для інженерної розробки поведінки роботів, а також в наочному встановленні структурних відповідностей між біологічними механізмами (реліз-стимул, мотиваційні змінні, ієрархія потреб) та їхніми інженерними аналогами (сенсорні тригери, поведінкові модулі, внутрішні змінні стану, механізми арбітражу дій). На відміну від наявних досліджень, які здебільшого розглядають окремі етологічні концепції ізольовано (наприклад, лише мотиваційні моделі або лише subsumption-архітектуру), у даній роботі пропонується наскрізний погляд на етологічний підхід як єдину методологічну основу – від елементарних поведінкових примітивів до соціальної взаємодії людина-робот у межах концепції ethorobotics.

Практична цінність дослідження полягає у можливості використання систематизованих етологічних принципів для проєктування реальних робототехнічних систем – сервісних, соціальних та мобільних роботів, – що потребують адаптивної, робастної та зрозумілої людині поведінки без необхідності побудови повної моделі середовища. Запропонований підхід може бути застосований при розробці архітектур управління поведінкою для роботів-асистентів, роботів для взаємодії з людьми похилого віку чи дітьми, а також автономних мобільних платформ, що функціонують у змінних умовах реального світу. Крім того, результати роботи можуть слугувати теоретичною основою для подальшої інтеграції етологічних моделей із сучасними методами машинного навчання, зокрема глибоким навчанням з підкріпленням.

Ключові етологічні концепції, найбільш продуктивні для інженерної реалізації у робототехніці є наступними:

1. **Поведінкові примітиви (fixed action patterns).** У класичній етології – це відносно жорсткі, генетично закріплені послідовності дій, які запускаються специфічним стимулом (реліз-стимулом). У робототехніці аналогом є елементарні поведінкові модулі (behaviour modules) – короткі, добре відпрацьовані алгоритми реагування на конкретний клас сенсорних подразників (уникнення перешкоди, слідування за об'єктом, повернення до станції заряджання).

2. **Мотиваційні змінні та внутрішні стани.** Поведінка тварини визначається не лише зовнішнім стимулом, а й внутрішнім мотиваційним станом (голод, втома, тривога). Це напряму переноситься в робототехніку через введення внутрішніх змінних (рівень заряду батареї, «втома» приводів, пріоритетність задачі), які модулюють поріг активації того чи іншого поведінкового модуля.

3. **Ієрархія та конкуренція поведінок (behaviour arbitration).** Етологічні моделі конкуренції мотивацій [2–4] дають теоретичне підґрунтя для архітектур типу *subsumption architecture* [5] та *action selection* – механізмів, які визначають, який з одночасно активних поведінкових модулів отримує контроль над актуаторами робота [6].

4. **Навчання через підкріплення в контексті адаптивної поведінки.** Аналогія з оперантним навчанням у тварин дозволяє інтегрувати етологічні поведінкові патерни з методами навчання з підкріпленням, забезпечуючи адаптацію ваг/порогів активації поведінкових модулів у процесі взаємодії з середовищем [7].

5. **Ethorobotics як прикладний напрям.** Окремо розглядається напрям *ethorobotics* (за аналогією з роботами А. Міклоші), орієнтований на моделювання соціальної та комунікативної поведінки роботів для природної взаємодії «людина – робот», зокрема на основі спостережень за поведінкою собак [8].

Інженерна реалізація ключових етологічних концепцій показана на наступних схемах (рис. 1–5)

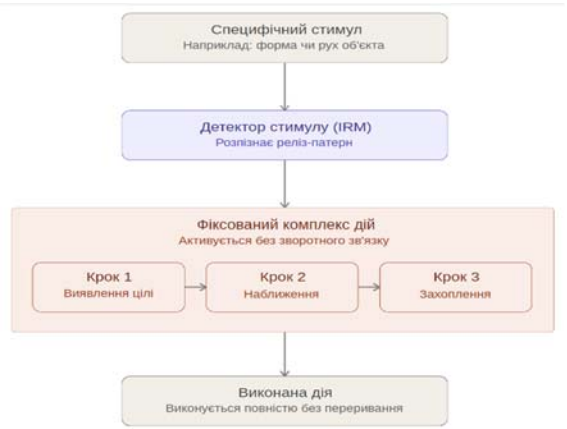


Рисунок 1. – Приклад інженерної реалізації концепції поведінкових примітивів

Схема на рис. 1 ілюструє інженерну реалізацію класичного етологічного механізму поведінкового примітива: специфічний реліз-стимул активує детектор стимулу, який запускає фіксований комплекс дій – жорстку послідовність із трьох кроків, що виконується повністю без переривання, незалежно від подальших змін у середовищі.



Рисунок 2. – Приклад інженерної реалізації концепції мотиваційних змінних та внутрішніх станів

Схема на рис. 2 ілюструє інженерну реалізацію етологічної концепції мотиваційних змінних та внутрішніх станів: зовнішній стимул і внутрішній стан робота (заряд батареї, «втома» приводів) об'єднуються у мотиваційну змінну, яка формує поріг активації – саме цей поріг визначає, чи достатньо сильний стимул, щоб «запустити» конкретний поведінковий модуль.



Рисунок 3. – Приклад інженерної реалізації концепції ієрархії та конкуренції поведінок

Схема на рис. 3 ілюструє конкуренцію трьох поведінкових модулів (уникнення перешкоди, слідування за ціллю, повернення на базу), кожен зі своїм рівнем пріоритету. Механізм арбітражу дій обирає, який модуль отримає контроль над актуаторами робота, пригнічуючи модулі з нижчим пріоритетом.



Рисунок 4. – Приклад інженерної реалізації концепції навчання через підкріплення

Схема на рис. 4 ілюструє цикл адаптивного навчання: поведінковий модуль виконує дію в середовищі, середовище генерує сигнал підкріплення – нагороду чи покарання, який оновлює пороги активації поведінкових модулів. Замкнений контур праворуч показує, що оновлені пороги впливають уже на наступний цикл вибору поведінки – це і є аналог оперантного навчання тварин, перенесений в архітектуру автономного робота.



Рисунок 5. – Приклад інженерної реалізації концепції Ethorobotics

Схема на рис. 5 ілюструє логіку концепції ethorobotics: спостереження за поведінкою тварини (наприклад, собаки) дає змогу виявити комунікативні патерни, які переносяться в модель соціальної поведінки робота, забезпечуючи природну, довірливу взаємодію людини з роботом без потреби у зовнішній людиноподібності самого робота [9].

Практична цінність підходу полягає у тому, що етологічно орієнтовані моделі:

– забезпечують реактивність та стійкість до непередбачуваних змін середовища;

– не потребують повної та точної моделі світу;

– легко масштабуються шляхом додавання нових поведінкових модулів без переробки всієї системи;

– дають природну основу для довірливої, зрозумілої людині взаємодії з роботом (соціальні роботи, роботи-асистенти).

Висновки. Етологічний підхід є продуктивною методологічною базою для розробки архітектур поведінки автономних роботів, особливо в задачах, що вимагають адаптивності, робастності та природності взаємодії з людиною. Перспективним напрямом подальших досліджень є формалізація мотиваційних моделей у вигляді обчислюваних функцій активації та їх інтеграція з сучасними методами машинного навчання (глибокого навчання з підкріпленням) для автоматичного налаштування поведінкових пріоритетів.

Список використаних джерел

1. Lorenz K. King Solomon’s Ring: New Light on Animal Ways. London : Methuen, 1952. P. 189.

2. Lorenz K. The Foundations of Ethology. Vienna : Springer-Verlag, 1981. P. 362.

3. Tinbergen N. The Study of Instinct. Oxford : Clarendon Press, 1951. P. 228.

4. Tinbergen N. On aims and methods of ethology. *Zeitschrift für Tierpsychologie*. 1963. Vol. 20, No. 4. P. 410–433.

5. Brooks R. A. A robust layered control system for a mobile robot. *IEEE Journal on Robotics and Automation*. 1986. Vol. 2, No. 1. P. 14–23.

6. Brooks R. A. Intelligence without representation. *Artificial Intelligence*. 1991. Vol. 47, No. 1–3. P. 139–159.

7. Arkin R. C. Behavior-Based Robotics. Cambridge, MA : MIT Press, 1998. P. 506.

8. Miklósi Á. Dog Behaviour, Evolution, and Cognition. Oxford : Oxford University Press, 2007. P. 377.

9. Miklósi Á., Korondi P., Matellán V. and Gácsi M. Ethorobotics: A New Approach to Human-Robot Relationship. *Frontiers in Psychology.* 2017. Vol. 8: 958. DOI: 10.3389/fpsyg.2017.00958

УДК 621.355:629.735-519

Микола Ткаченко
старший викладач кафедри фізичного виховання
та основ національного спротиву
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ОБҐРУНТУВАННЯ КРИТЕРІЇВ ВИБОРУ ТА ЕКСПЛУАТАЦІЇ ЕЛЕМЕНТІВ ЖИВЛЕННЯ ДЛЯ БЕЗПІЛОТНИХ ЛІТАЛЬНИХ АПАРАТІВ

Актуальність. Ще десять років тому безпілотник був радше іграшкою або вузькоспеціальним інструментом. Сьогодні це масовий виріб: FPV-дрони виготовляють великими серіями, з'явилися перехоплювачі, а літаки-розвідники працюють на відстані в сотні кілометрів. Проте конструктор будь-якого з цих апаратів рано чи пізно впирається в те саме обмеження – батарею. Від неї залежить, скільки апарат протримається в повітрі, який вантаж підніме і чи вистачить йому тяги на різкий маневр [1, 2]. Попри це, акумулятор часто беруть «як у сусіда» або найдешевший із наявних. Чи підходить він до двигунів і регуляторів, з'ясовують уже в польоті: напруга провалюється на повному газі, гріється електроніка, а після двадцяти–тридцяти вильотів елементи здуваються. Простих і водночас обґрунтованих критеріїв вибору, якими міг би користуватися не лише досвідчений інженер, бракує.

Мета даної роботи – з'ясувати, на які параметри акумулятора треба дивитися передусім, порівняти між собою літієві батареї різних типів і скласти покрокову схему, за якою джерело живлення можна підібрати під конкретне завдання БПЛА.

Методи та матеріали. Було порівняно п'ять типів акумуляторів, які реально використовують у безпілотниках: Li-Po, Li-HV, циліндричні Li-Ion форматів 18650 і 21700, пакетні Li-NMC та

LiFePO4. Зосереджено увагу на шести показниках: питому енергію, C-рейтинг (тобто допустимий струм розряду), напругу одного елемента, кількість циклів до помітної деградації, безпечність і роботу на морозі. Цифри взяті з паспортів виробників та з оглядових статей [1–4]. Час польоту розраховано за простою моделлю: ємність ділиться на середній струм, а результат зменшується з огляду на те, що розряджати батарею до критично низького рівня не можна.

Порівняння типів акумуляторів. Зведені дані подано в таблиці 1.

Таблиця 1. – Порівняльні характеристики акумуляторів для БПЛА

Параметр	**Li-Po**	**Li-HV**	**Li-Ion (18650, 21700)**	**Li-NMC (пакетні)**	**LiFePO4**
Питома енергія, Вт·год/кг	160-200	180-230	220-280	240-300	120-160
Струмовіддача (C-рейтинг)	30-100C і більше	50-120C і більше	3-10C	5-15C	10-30C
Напруга елемента номінальна / максимальна, В	3,7 / 4,2	3,8 / 4,35	3,6 / 4,2	3,6 / 4,2	3,2 / 3,65
Ресурс до 80% ємності, циклів	150-300	100-200	300-500	500-800	2000-5000
Термічна стійкість і вибухобезпечність	низька	низька, чутливі до перезаряду	середня	середня	висока
Просідання напруги під навантаженням	мінімальне	мінімальне	помітне	середнє	мале, «пласка» розрядна крива
Робота за температури –20 °C	значна втрата ємності	значна втрата ємності	посередня, бажаний підігрів	задовільна	знижена, потрібен підігрів

Примітка. Наведено типові діапазони значень; характеристики конкретних виробів залежать від виробника та серії

Головне, що відображено в таблиці: ідеальної батареї не існує. Той елемент, що добре віддає струм – мало запасає; той, що служить тисячі циклів – важкий. Li-Po і Li-HV – лідери за струмовіддачею, напруга в них під навантаженням майже не просідає. Через це на них літають спортивні, ударні FPV і перехоплювачі; типова акумуляторна збірка – від 4S до 6S. У Li-HV елемент заряджають до 4,35В замість звичних 4,2В, тож за тієї самої маси він дає трохи більше енергії. Недолік –

мала кількість циклів заряду та розряду та підвищена чутливість до перезаряду.

Наочно це видно на рисунку 1: у координатах «питома енергія – струмовіддача» високострумові батареї зібралися вгорі, енергоємні – праворуч унизу, а правий верхній кут, де мала б бути «ідеальна» батарея, залишається порожнім.

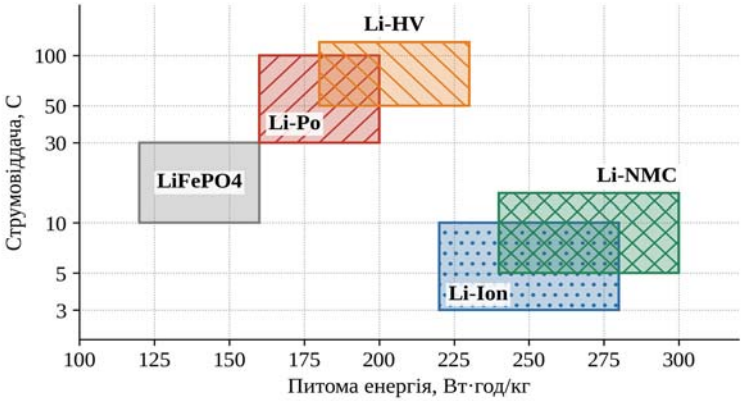


Рисунок 1. – Типи акумуляторів у координатах «питома енергія – струмовіддача» (побудовано за даними таблиці 1)

З циліндричними Li-Ion та пакетними Li-NMC ситуація зворотна. Енергії на кілограм у них більше, а от різко віддати великий струм вони не можуть. Тому їх використовують там, де навантаження рівне: на дрони далекого радіусу дії та літальні крила з часом польоту 20-40 хв і більше.

Окремо стоїть LiFePO4. За питомою енергією це найслабший варіант із п’яти, але ресурс у нього 2000-5000 циклів, а термічна стійкість найвища: пробитий уламком елемент загоряється значно рідше, ніж Li-Po. Звідси й сфери, де маса не головне. Це наземні станції керування і ретранслятори. Це буферні батареї в гібридних установках літакових БПЛА, де основну енергію дає двигун внутрішнього згоряння з генератором. Це вантажні та моніторингові дрони, які літають по 5-10 разів на добу і мають швидко заряджатися. Нарешті, це навчальні апарати: там Li-Po збірки зношуються швидко через малу кількість циклів заряду та розряду, а міняти їх дорого. Є і слабке місце – холод. На морозі LiFePO4 відчутно втрачає ємність,

заряджати його за мінусової температури не можна, тому взимку без підігріву не обійтися.

Параметри сумісності. Коли тип батареї обрано, її треба узгодити із силовою установкою. Перевіряють три речі.

Напруга. Число послідовних елементів (S) мусить вкладатися в діапазон, на який розраховані регулятори обертів (ESC), польотний контролер і двигуни. Завищена напруга пошкодить електроніку. Занижена – залишить апарат без тяги й змусить силову частину працювати на більших струмах.

Струмовіддача. Найбільший струм, який батарея здатна віддавати тривало, знаходять так:

$$I_{max} = C \cdot Q, \tag{1}$$

де:

I_{max} – максимальний струм розряду, А;

C – С-рейтинг акумулятора, год$^{-1}$;

Q – ємність, А·год.

Скажімо, батарея на 1,5 А·год із рейтингом 100С за паспортом дає 150 А. П'ятидюймовому спортивному FPV-дрону потрібні рейтинги порядку 80–120С, бо на слабшій батареї напруга на піках просідають. Варто пам'ятати й про те, що виробники свої рейтинги нерідко завищують, отже запас за струмом зайвим не буде.

Маса та роз'єми. Батарея не повинна бути заважкою для гвинтомоторної групи; для п'ятидюймового апарата нормальною вважають масу 150–220 г. Роз'єм (BT2.0, XT30 чи XT60) беруть такий, що тривало витримує робочий струм і не гріється.

Розрахунок тривалості польоту. Для оцінки часу польоту користуємося формулою

$$t = (Q / I) \cdot 60 \cdot k, \tag{2}$$

де:

t – тривалість польоту, хв;

Q – ємність акумулятора, мА·год;

I – середній струм споживання, мА;

k – коефіцієнт використання ємності.

За звичайних умов для літієвих батарей беруть k = 0,8–0,9, тобто 10–20 % ємності свідомо не використовують, аби не розрядити

елементи надто глибоко. Якщо батарея стара або надворі мороз, k краще зменшити до 0,7.

Перевіримо формулу на двох прикладах. Li-Po на 1500 мА·год, середній струм 22 А (22 000 мА), k = 0,85: виходить $t \approx 3{,}5$ хв. Приблизно стільки й літає п'ятидюймовий FPV-дрон в активному режимі. Li-Ion збірка на 8000 мА·год за струму 12 А дає вже близько 34 хв. Різниця майже вдесятеро, і зумовлена вона не тільки ємністю: другий апарат від початку розрахований на спокійний економний політ.

Послідовність добору акумулятора. Усе наведене можна звести до шести кроків:

1) Описати завдання: скільки треба літати, яким буде навантаження (ривками чи рівне), за якої температури;

2) Під цей характер навантаження вибрати тип батареї за таблицею 1;

3) Підібрати число елементів S так, щоб напруга підходила регуляторам і двигунам;

4) Узяти максимальний струм силової установки, додати 20-30% запасу і за формулою (1) перевірити, чи вистачає C-рейтингу;

5) За формулою (2) знайти ємність під потрібний час польоту і подивитися, чи не вийшла батарея заважкою;

6) Переконатися у тому, що роз'єми та проводи розраховані на цей струм, і заздалегідь продумати, як батарею заряджати й зберігати.

Перспективи розвитку. Розвиток джерел живлення для БПЛА йде одразу кількома шляхами. Перший – кремній-вуглецеві аноди. З ними питома енергія Li-Ion елементів може зрости з теперішніх 250-280 до 350-400 Вт·год/кг, а це на 30-50% довший політ у тих самих габаритах. Другий – твердотільні акумулятори: рідкий електроліт у них замінено керамікою або полімером, очікувана питома енергія становить 400-500 Вт·год/кг, а ймовірність займання після пошкодження набагато нижча [5].

Частина змін відбувається вже зараз. Елементи формату 21700 навчилися віддавати великі струми: Molicel P45B за ємності 4,5 А·год тривало тримає 45 А [4]. Раніше таке було під силу лише Li-Po, а тепер Li-Ion збірки ставлять і на важкі FPV-дрони. Розумнішають і системи керування батареєю (BMS). По шині CAN або SMBus вони повідомляють польотному контролеру рівень заряду (SoC) і ступінь зношення (SoH) кожного елемента, а вбудований підігрів дає змогу злітати взимку без помітної втрати ємності в перші хвилини. Якщо ж апарат має триматися в повітрі 10-24 год, одного акумулятора замало. Тут перспективні гібриди, в яких генератор на двигуні внутрішнього

згоряння чи водневий паливний елемент працює разом із буферною батареєю LiFePO4 або Li-NMC [1, 3].

Висновки. 1. Акумулятор для БПЛА завжди вибирають із компромісом: питома енергія, струмовіддача, ресурс і безпечність одночасно високими не бувають.

2. Там, де навантаження пікове (гоночні й ударні FPV-дрони, перехоплювачі), найкраще працюють Li-Po та Li-HV. Для довгих польотів із рівним навантаженням більше підходять Li-Ion і Li-NMC. LiFePO4 варто залишити для наземної техніки, буферних накопичувачів та апаратів, що літають багато разів на день.

3. Двох простих формул і шести кроків добору достатньо, щоб ще до закупівлі з'ясувати, чи сумісна батарея із силовою установкою і скільки апарат реально пролітає.

4. У найближчі роки на характеристики БПЛА найбільше вплинуть кремній-вуглецеві аноди, твердотільні елементи, високострумові Li-Ion, «розумні» BMS і гібридні силові установки.

Список використаних джерел

1. Коршунова О., Комар М. Інформаційна технологія повітряного старту БПЛА. *Наука і техніка сьогодні.* 2025. № 11(52). С. 2325–2337. DOI: 10.52058/2786-6025-2025-11(52)-2325-2337

2. Pandey A. P., Yadav J. K., Dixit A. LiFePO4 as a cathode material. Advancement in oxide utilization for li rechargeable batteries. 2025. P. 318–347. DOI: 10.1039/9781837673612-00318

3. Marcisz M., Kozuba J., Ulman K. Development directions of energy sources for unmanned aerial vehicle (UAV). *Scientific Journal of Silesian University of Technology. Series Transport.* 2024. Vol. 125. P. 177–189. DOI: 10.20858/sjsutst.2024.125.12

4. INR-21700-P45B. Lithium-ion rechargeable battery : product data sheet / E-One Moli Energy Corp. URL: https://www.imrbatteries.com/content/molicel_p45b.pdf (дата звернення: 05.10.2026).

5. Pandey A. P., Yadav J. K., Dixit A. LiFePO4 as a cathode material. Advancement in oxide utilization for li rechargeable batteries. 2025. P. 318–347. DOI: 10.1039/9781837673612-00318 (дата звернення: 15.09.2026).

УДК 004.056.5

Єгор Ухань
PhD, в.о. викладача
кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

МЕТОДИ ТА ЗАСОБИ ЗАБЕЗПЕЧЕННЯ ЗАХИЩЕНОГО МЕРЕЖЕВОГО З'ЄДНАННЯ НА БАЗІ ТЕХНОЛОГІЙ WIREGUARD І OPENVPN

Вступ. Розвиток технологій віддаленого доступу, територіально розподілених корпоративних мереж і хмарних обчислень зумовлює підвищення вимог до захисту інформації під час її передавання через загальнодоступні мережі. Одним із засобів забезпечення конфіденційності, цілісності та автентичності мережевого обміну є технології віртуальних приватних мереж (VPN). Серед поширених програмних рішень виділяються OpenVPN, що використовує механізми TLS, та WireGuard, який характеризується спрощеною архітектурою і підтримкою в ядрі Linux [1–2]. Вибір відповідної технології визначається вимогами до безпеки, продуктивності, керованості та сумісності з наявною мережевою інфраструктурою. Питання вибору й порівняльного оцінювання VPN-технологій для корпоративних мереж розглянуто в працях [3–4].

Мета даної роботи полягає в обґрунтуванні архітектури захищеної мережевої інфраструктури на основі WireGuard та OpenVPN шляхом порівняльного аналізу механізмів захисту, протокольних накладних витрат і експлуатаційних характеристик зазначених технологій.

Особливості протоколів. Архітектура WireGuard характеризується мінімізацією кількості компонентів і криптографічних механізмів. Протокол побудований на Noise Protocol Framework (шаблон Noise_IKpsk2) [5], використовує Curve25519 для обміну ключами, ChaCha20-Poly1305 для автентифікованого шифрування [6] та BLAKE2s для хешування. Кожен вузол ідентифікується статичним відкритим ключем, а список дозволених підмереж (AllowedIPs) одночасно виконує роль таблиці маршрутизації та простої політики фільтрації. Сесійні ключі оновлюються приблизно кожні дві хвилини за наявності трафіку, що сприяє забезпеченню властивості прямої таємності (forward secrecy). Протокол став предметом формального криптографічного аналізу [7]. Вузол не відповідає на пакети від

невідомих відправників, що знижує можливість виявлення вузла за допомогою стандартного сканування портів. До обмежень належать робота лише поверх UDP та відсутність вбудованих механізмів видачі адрес і централізованого відкликання доступу.

OpenVPN використовує TLS, зокрема TLS 1.3 [8], для автентифікації сторін і узгодження ключів, а дані каналу передає у власному форматі поверх UDP або TCP. Підтримка сертифікатів X.509 дозволяє побудувати повноцінну інфраструктуру відкритих ключів з відкликанням сертифікатів (CRL), а плагіни – додати парольну та багатофакторну автентифікацію. Використання TCP-порту 443 може підвищувати сумісність із мережами, у яких застосовуються сувора фільтрація трафіку. До обмежень належать більший обсяг програмної реалізації (близько 70 тис. рядків проти близько 4 тис. у WireGuard), обробка пакетів переважно в просторі користувача, що збільшує кількість перемикань контексту, та складніша конфігурація. Порівняльну характеристику технологій наведено в табл. 1.

Таблиця 1. – Порівняльна характеристика WireGuard та OpenVPN

Характеристика	**WireGuard**	**OpenVPN**
Автентифікація	Статичні відкриті ключі вузлів (Curve25519)	Сертифікати X.509, логін і пароль, OTP через плагіни
Криптографічні примітиви	Noise_IKpsk2, ChaCha20-Poly1305, BLAKE2s	TLS 1.2/1.3, AES-GCM, ChaCha20-Poly1305
Транспорт	Лише UDP	UDP та TCP (зокрема порт 443)
Реалізація	Модуль ядра Linux (з версії 5.6)	Переважно процес у просторі користувача
Кодова база	Близько 4 тис. рядків	Значно більша, залежить від TLS-бібліотеки
Оновлення ключів	Автоматично, приблизно кожні 2 хв за наявності трафіку	За налаштуванням (типово щогодини)

Характеристика	WireGuard	OpenVPN
Керування клієнтами	Статична конфігурація AllowedIPs	Динамічна видача адрес, push-параметри, CRL

Протокольні накладні витрати. Ефективність передавання даних через VPN-тунель залежить, зокрема, від обсягу службових даних, що додаються під час інкапсуляції пакетів. Для оцінювання максимально допустимого розміру вкладеного пакета використовується співвідношення між MTU фізичного каналу та сумарним розміром заголовків мережевого, транспортного й VPN-протоколів (1):

$$MTU_{tun} = MTU_{link} - (H_{IP} + H_{UDP} + H_{VPN}) \quad (1)$$

де MTU_{link} – MTU фізичного каналу; H_{IP} – розмір заголовка IP (20 байт для IPv4, 40 байт для IPv6); H_{UDP} – заголовок UDP (8 байт); H_{VPN} – службова частина протоколу VPN.

Частку корисних даних у пакеті визначає формула (2):

$$\eta = \frac{P}{P + H_{IP} + H_{UDP} + H_{VPN}} \cdot 100\ \% \quad (2)$$

де P – розмір вкладеного пакета.

Для WireGuard H_{VPN} = 32 байти (16 байт заголовка й 16 байт мітки автентифікації), для OpenVPN у режимі UDP з AES-GCM – близько 24 байт. За P = 1400 байт та IPv4 отримуємо $\eta \approx 95{,}9\%$ для WireGuard і $\approx$ 96,4% для OpenVPN, тобто різниця за рахунок заголовків не перевищує одного відсотка. Для WireGuard при IPv6 та каналі з MTU 1500 байт формула (1) дає $1500 - (40 + 8 + 32) = 1420$ байт, що збігається зі значенням, яке використовується за замовчуванням. Таким чином, за розглянутих параметрів різниця в частці корисного навантаження, зумовлена протокольними накладними витратами, є незначною. Відмінності у фактичній пропускній здатності можуть визначатися архітектурою програмної реалізації, характеристиками апаратної платформи, вибором криптографічних алгоритмів і параметрами конфігурації тунелю.

Запропонована архітектура. Архітектуру захищеної мережевої інфраструктури сформовано відповідно до топології типу «зірка» (рис. 1). Центральний шлюз під керуванням Linux одночасно обслуговує обидві технології. WireGuard передбачається використовувати для

організації міжвузлових з'єднань між філіями та центральним вузлом, а також для адміністративного доступу. Такий вибір обґрунтовується відносною простотою конфігурації протоколу, особливостями керування криптографічними ключами та вимогами до продуктивності мережевих з'єднань. OpenVPN доцільно застосовувати для організації віддаленого доступу користувачів за умов, коли необхідно централізовано керувати сертифікатами, реалізовувати додаткові механізми автентифікації та забезпечувати сумісність із мережами, у яких діють обмеження на передавання трафіку.

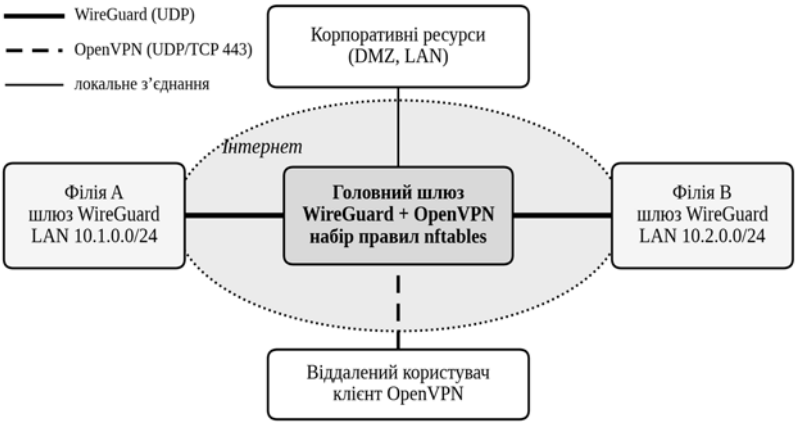


Рисунок 1. – Архітектура захищеної мережі на основі WireGuard та OpenVPN

Додатково передбачено: окрему пару ключів для кожного вузла з суворим обмеженням прав доступу до приватних ключів і регламентною ротацією; використання параметра PresharedKey у WireGuard як додаткового симетричного шару захисту, зокрема на випадок майбутніх квантових атак на Curve25519; застосування tls-crypt та мінімальної версії TLS 1.2 в OpenVPN; обмеження частоти підключень, централізований збір журналів і моніторинг стану тунелів; резервування шлюзу.

Модель загроз. Для оцінювання захищеності запропонованої мережевої архітектури визначено основні загрози, пов'язані з передаванням даних через загальнодоступні мережі. До них належать перехоплення та модифікація трафіку, повторне відтворення пакетів, компрометація криптографічних ключів, спроби несанкціонованого

доступу, атаки на доступність сервісів і втрата пристроїв користувачів. Відповідність основних механізмів захисту зазначеним загрозам наведено в табл. 2. Обидві технології забезпечують криптографічний захист передавання даних, однак відрізняються підходами до керування ключами та відкликання доступу.

Таблиця 2. – Загрози та механізми захисту

Загроза	**WireGuard**	**OpenVPN**
Перехоплення трафіку	Шифрування ChaCha20-Poly1305	Шифрування AES-GCM або ChaCha20-Poly1305 у каналі даних
Модифікація та повторне відтворення пакетів	Автентифікація кожного пакета, лічильник із ковзним вікном	Мітка автентифікації та порядковий номер пакета
Компрометація довгострокового ключа	Ефемерні ключі сесії, пряма таємність	Ефемерний обмін ключами в TLS (ECDHE)
Сканування портів і перевантаження	Мовчання щодо невідомих відправників, cookie-механізм під навантаженням	HMAC-захист керуючого каналу (tls-crypt, tls-auth)
Втрата або крадіжка пристрою клієнта	Видалення відкритого ключа з конфігурації шлюзу	Відкликання сертифіката (CRL), додатковий фактор OTP

Експлуатаційні аспекти. Під час експлуатації VPN-інфраструктури необхідно враховувати низку технічних чинників. Для вузлів за NAT у WireGuard використовується параметр PersistentKeepalive (зазвичай 25 с), який підтримує відображення в таблиці трансляції адрес. Щоб уникнути фрагментації, доцільно задати значення MTU тунельного інтерфейсу за формулою (1) та обмежити MSS для TCP-з’єднань на шлюзі. Передбачено маршрутизацію DNS-запитів через захищений тунель, щоб запобігти витоку імен, та правило блокування трафіку поза тунелем (kill switch). Стан тунелів контролюється за часом останнього рукостискання (WireGuard) та журналами підключень (OpenVPN), а розгортання конфігурацій вузлів доцільно автоматизувати, наприклад Ansible, що знижує ризик

помилок. Для підвищення відмовостійкості центрального шлюзу передбачено використання двох серверів із механізмом VRRP та спільною віртуальною адресою.

Місце серед інших рішень. Окрім WireGuard та OpenVPN, для побудови захищених мережевих з'єднань застосовується IPsec із використанням IKEv2 для узгодження параметрів безпеки. До переваг цього підходу належать підтримка в багатьох операційних системах і мережевих пристроях, а також можливість взаємодії обладнання різних виробників. Водночас складність конфігурації залежить від реалізації та кількості параметрів, які необхідно узгодити. Вибір між WireGuard, OpenVPN та IPsec доцільно здійснювати з урахуванням вимог до керування доступом, сумісності, продуктивності й особливостей мережевого середовища.

Методика перевірки. Для порівняльного оцінювання WireGuard та OpenVPN передбачається використання тестового стенда з однаковими апаратними й мережевими умовами. Пропускну здатність каналу планується вимірювати за допомогою утиліти iperf3, затримку передавання пакетів – за допомогою ping, а рівень використання процесорних ресурсів – засобами моніторингу операційної системи. Для забезпечення зіставності результатів необхідно використовувати однакові профілі трафіку, узгоджені параметри MTU та зіставні криптографічні налаштування. Втрати пропускної здатності, пов'язані з використанням VPN-тунелю, оцінюються шляхом порівняння показників прямого з'єднання та з'єднання через тунель за формулою (3):

$$K = \frac{T_{\text{пр}} - T_{vpn}}{T_{\text{пр}}} \cdot 100\,\% \qquad (3)$$

де $T_{пр}$ – пропускна здатність прямого з'єднання без тунелю;
T_{vpn} – пропускна здатність через тунель.

Висновки. На основі порівняльного аналізу WireGuard та OpenVPN визначено відмінності їхніх архітектурних підходів, механізмів автентифікації, керування криптографічними ключами та організації віддаленого доступу. Обґрунтовано доцільність використання комбінованої VPN-архітектури, у якій WireGuard застосовується для міжвузлових з'єднань і адміністративного доступу, а OpenVPN – для сценаріїв, що потребують централізованого керування сертифікатами та додаткових механізмів автентифікації. Визначено основні загрози для захищених мережевих з'єднань і систематизовано відповідні механізми протидії. Подальші дослідження доцільно спрямувати на

експериментальне оцінювання пропускної здатності, затримки передавання даних і використання обчислювальних ресурсів на різних апаратних платформах.

Список використаних джерел

1. OpenVPN Community Resources. Reference Manual for OpenVPN 2.6. URL: https://openvpn.net/community-resources/ (дата звернення: 01.10.2026).
2. Donenfeld J. A. WireGuard: Next Generation Kernel Network Tunnel. Proceedings of the Network and Distributed System Security Symposium (NDSS). 2017. URL: https://www.ndss-symposium.org/wp-content/uploads/2017/09/ndss2017_04A-3_Donenfeld_paper.pdf (дата звернення: 08.10.2026).
3. Тишик І. Вибір технології віддаленого доступу для ефективної організації захисту мережевих з'єднань. *Кібербезпека: освіта, наука, техніка.* 2023. Том 3, № 19. С. 34–45. DOI: 10.28925/2663-4023.2023.19.3445
4. Фомін В. В., Покотило О. А. Аналіз технологій VPN для використання у корпоративному середовищі : *тези VII Всеукраїнської науково-технічної конференції «Комп'ютерні технології: інновації, проблеми, рішення»,* м. Житомир, 02-03 грудня 2024 р. Житомир : Житомирська політехніка, 2024. URL: https://conf.ztu.edu.ua/wp-content/uploads/2025/01/povnyj-tekst-1.pdf (дата звернення: 08.10.2026).
5. Perrin T. The Noise Protocol Framework. 2018. Revision 34. URL: https://noiseprotocol.org/noise.html (дата звернення: 18.10.2026).
6. Nir Y., Langley A. ChaCha20 and Poly1305 for IETF Protocols: RFC 8439. Internet Engineering Task Force, 2018. URL: https://www.rfc-editor.org/info/rfc8439 (дата звернення: 11.10.2026).
7. Dowling B., Paterson K. G. A Cryptographic Analysis of the WireGuard Protocol. Applied Cryptography and Network Security (ACNS 2018): Lecture Notes in Computer Science. Cham : Springer, 2018. Vol. 10892. URL: https://eprint.iacr.org/2018/080.pdf (дата звернення: 08.10.2026).
8. Rescorla E. The Transport Layer Security (TLS) Protocol Version 1.3: RFC 8446. Internet Engineering Task Force, 2018. URL: https://www.rfc-editor.org/info/rfc8446 (дата звернення: 08.10.2026).

УДК 004.056

Євгеній Худолій
аспірант кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євген Дарнапук
PhD, доцент,
в.о. завідувача кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

РОЛІ ТА ПОВНОВАЖЕННЯ ВИДАВЦІВ ПЕРЕВІРЮВАНИХ ОБЛІКОВИХ ДАНИХ У МІЖДОМЕННІЙ АВТОРИЗАЦІЇ

У багатохмарній комп'ютерній системі рішення про надання сервісу доступу до ресурсу іншої організації може спиратися на відомості від кількох видавців, кожен із яких засвідчує певні твердження. Наприклад, організація, що керує сервісом, підтверджує його належність до свого адміністративного домену, тоді як власник запитуваного ресурсу визначає те, чи має сервіс право виконати конкретну операцію. Однак повноваження підтверджувати належність сервісу самі по собі не дають видавцеві права засвідчувати доступ до чужого ресурсу [1]. Тому сторона, яка ухвалює рішення про доступ, має визначити, які твердження вона приймає від кожного видавця та до яких суб'єктів, ресурсів і операцій вони застосовуються. Це зумовлює потребу розмежувати ролі видавців і встановити межі використання їхніх тверджень у міждоменній авторизації.

Перевірювані облікові дані дають змогу подавати твердження видавця у формі, яку можна перевірити криптографічними засобами. Однак така перевірка не підтверджує достовірності всіх тверджень, закодованих в облікових даних. У поясненні моделі довіри W3C окремо зазначено, що вибір видавців, яким довіряє верифікатор, а також визначення даних і цілей цієї довіри залишаються поза межами специфікації. Для того, щоб використовувати перевірювані облікові дані для авторизації, потрібен відповідний механізм керування доступом [2]. У міждоменному сценарії необхідно визначити допустимі повноваження для кожного джерела тверджень. Для маркування сервісів та видавців можна використовувати децентралізовані ідентифікатори, які мають властивість

assertionMethod у документах, і визначають методи перевірки для засвідчення тверджень, зокрема видавання перевірюваних облікових даних [3]. У цій задачі слід розмежовувати допустимість використання ключа для засвідчення твердження та визнане стороною ресурсу право видавця засвідчувати конкретний атрибут або дозвіл, оскільки сама наявність децентралізованого ідентифікатора не надає такого права.

Сценарій доступу та вихідні умови. Розглянемо взаємодію двох організацій у спільному проєкті. Організація А розгортає у своєму хмарному середовищі аналітичний сервіс S, який обчислює статистику на основі даних вимірювань. Організація Б зберігає набір цих даних D та надає доступ до нього через програмний інтерфейс. Організації утворюють незалежні адміністративні домени. Сторона Б дозволяє читання D лише сервісу з чинним підтвердженням належності до А та погодженим правом на цю операцію. Сервіс позначається власним децентралізованим ідентифікатором.

До початку обміну адміністратор Б налаштовує локальну конфігурацію довіри. Вона пов’язує ідентифікатор видавця А з повноваженням підтверджувати належність сервісів до А, а ідентифікатор видавця Б з повноваженням підтверджувати право на читання D. Під областю повноважень видавця тут розуміємо дозволені типи тверджень і межі їх застосування до суб’єктів, ресурсів та операцій. Видавець не може розширити цю область, додавши нове твердження до власних перевірених облікових даних. Підготовка до доступу передбачає дві окремі дії. Уповноважена служба А реєструє сервіс S і надає йому належність до облікових даних. Власник набору D погоджує читання для цього сервісу, після чого уповноважена служба Б надає обліковим даним права доступу. Призначення та зміст цих перевірених облікових даних наведено в табл. 1.

Таблиця 1. – Призначення та зміст перевірюваних облікових даних у розглянутому сценарії

Призначення облікових даних	**Видавець і підстава видачі**	**Зміст твердження**
Підтвердження належності	Уповноважена служба А після реєстрації сервісу та підтвердження його належності до організації	Сервіс із визначеним ідентифікатором належить до організації А
Підтвердження права доступу	Уповноважена служба Б після погодження доступу власником набору D	Сервісу з тим самим ідентифікатором надано право читання набору D

Обидва типи перевірюваних облікових даних містять ідентифікатор, відомості про видавця, ідентифікатор суб'єкта, строк чинності, відомості для перевірки статусу та криптографічне підтвердження видавця. У облікових даних права доступу додатково вказуються ідентифікатор набору D та дозволена операція read. Значення цих прикладних атрибутів узгоджуються між сторонами. Видача права доступу потребує окремого рішення Б. Сервіс S зберігає отримані облікові дані та власний закритий ключ. Обидва набори перевірюваних облікових даних стосуються одного сервісу, S одночасно є його суб'єктом, тримачем і надавачем. Для наступних запитів використовуються ті самі облікові дані, доки вони залишаються чинними та прийнятними відповідно до політики Б.

Правило доступу. Політику доступу описано за підходом керування доступом на основі атрибутів (ABAC). Відповідно до NIST SP 800-162, рішення в ABAC ґрунтується на зіставленні атрибутів суб'єкта, об'єкта, запитуваної операції та, за потреби, умов середовища з політикою [4]. У розглянутому випадку враховуються децентралізовані ідентифікатори і належність сервісу, ідентифікатор ресурсу D, операція читання та підтверджене право на її виконання. Перед оцінюванням політики перевіряють походження й прийнятність тверджень, на підставі яких отримано ці відомості. Таким чином, ABAC визначає спосіб опису умов доступу, а перевірювані облакові дані визначають спосіб подання засвідчених відомостей для їх перевірки.

Політика Б дозволяє читання D, якщо одночасно виконано такі умови:

– належність S до A підтверджена прийнятними обліковими даними видавця A;

– право S на читання D підтверджене прийнятними обліковими даними видавця Б;

– ресурс і операція в запиті відповідають засвідченому праву.

Передумовою є успішна перевірка подання та його зв'язку з поточним запитом. За невиконання хоча б однієї необхідної умови або неможливість її перевірити доступ не надається. Компонент захисту програмного інтерфейсу реалізує це рішення.

Послідовність одного звернення до ресурсу:

1. Сервіс S надсилає стороні Б запит на читання набору D для обчислення статистики.

2. Б створює одноразове випадкове значення (англ. nonce) та зберігає для нього заявлений децентралізований ідентифікатор сервісу, ідентифікатор D, операцію read, строк відповіді та ознаку використання. Значення разом із параметрами запиту повертається через захищене з'єднання до автентифікованої сторони Б.

3. Сервіс зіставляє отримані параметри зі своїм запитом і формує перевірені подання, що містить інформацію про облікові дані. Для криптографічного підтвердження подання ключем сервісу передбачено Data Integrity: у challenge записується отримане одноразове значення, а в domain – очікуване позначення сторони Б. Специфікація визначає ці поля та перевірки їхніх очікуваних значень [5]. Сервіс надсилає перевірене подання через захищене з'єднання.

4. Верифікатор Б перевіряє криптографічні підтвердження перевіреного подання та вкладених облікових даних, а також призначення ключів, строки чинності та статуси облікових даних. Він звіряє domain з очікуваним значенням, а challenge – із збереженим записом. Децентралізований ідентифікатор, ресурс і операція мають відповідати запиту, тоді як строк відповіді не має спливти, а challenge не повинен бути використаним. Після отримання відповіді challenge позначається як використаний. Ідентифікатор суб'єкта обох облікових записів має збігатися з ідентифікатором сервісу, володіння ключем якого підтверджується поданням. Твердження видавців перевіряються за попередньо налаштованою конфігурацією довіри.

5. Прийняті відомості зіставляються з наведеним правилом доступу. За виконання всіх умов компонент захисту програмного інтерфейсу дозволяє читання D. Сервіс отримує набір D і обчислює статистику. Якщо необхідна умова не виконана або її неможливо перевірити, дані не передаються.

Видачу перевірених облікових даних та послідовність їх використання під час звернення показані на рис. 1.

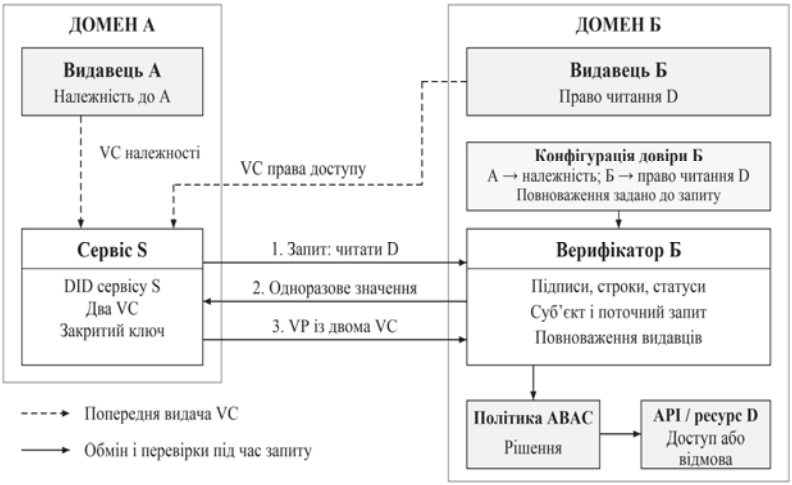


Рисунок 1. – Видача перевірюваних облікових даних (VC) та їх використання для міждоменної авторизації

Приклад структури перевірюваних облікових даних. Зміст облікових даних права доступу ілюструє наведений нижче скорочений фрагмент JSON за структурою VC Data Model 2.0 [2]. Умовний видавець Б засвідчує право сервісу S на читання набору D протягом визначеного строку.

```
{
  "id": "https://b.example/credentials/42",
  "type": ["VerifiableCredential",
"DatasetAccessCredential"],
  "issuer": "did:example:issuer-b",
  "validFrom": "2026-10-15T00:00:00Z",
  "validUntil": "2026-10-31T00:00:00Z",
  "credentialSubject": {
    "id": "did:example:service-s",
    "resource": "https://b.example/datasets/D",
    "action": "read"
  }
}
```

У прикладі issuer ідентифікує видавця, а credentialSubject.id – сервіс, якому надано право. DatasetAccessCredential, resource та action є прикладними розширеннями для цього сценарію, їхні значення потрібно визначити в контексті @context. Для стислості не наведено @context, credentialStatus і proof, тому цей фрагмент не є повним перевірюваним обліковим записом, придатним для криптографічної перевірки. Усі записи й адреси в прикладі умовні.

Аналіз відмов. Розглянемо випадок виходу видавця за межі повноважень. Сервіс подає чинні перевірювані облікові дані належності від А та додаткові облікові дані права читання D, також видані А. Обидва криптографічні підтвердження можуть бути коректними, однак конфігурація Б визнає за видавцем А лише повноваження щодо належності. Твердження про право читання не приймається, тому обов'язкова умова політики залишається невиконаною. Інший випадок: якщо А відкликає відповідні облікові дані, чинні права доступу від Б не забезпечують виконання обох умов політики. Новий запит відхиляється після встановлення факту відкликання. Аналогічно, облікові дані для іншого ідентифікатора, ресурсу чи операції не підтверджують право на поточний запит. Поведінка залежить від доступності та актуальності інформації про статус облікових записів. Якщо неможливо встановити потрібний статус установи, за визначеним правилом доступ не надається.

Висновок. Отже, у розглянутому сценарії розмежовано підтвердження належності сервісу та його права на ресурс. Визначено зміст двох типів перевірюваних облікових даних, порядок їх подання та умови застосування локальної політики ABAC. Повноваження видавця зіставляються зі змістом кожного твердження, а засвідчене право – із суб'єктом, ресурсом і операцією поточного запиту. Наведені випадки відмов є логічним аналізом сценарію, а не результатами експерименту. Їх перевірка на прототипі, а також час поширення відкликання й витрати на перевірки є завданнями для подальших досліджень.

Список використаних джерел

1. Rose S., Borchert O., Mitchell S., Connelly S. Zero Trust Architecture. NIST Special Publication 800-207. Gaithersburg : NIST, 2020. DOI: 10.6028/NIST.SP.800-207

2. Verifiable Credentials Data Model v2.0. W3C Recommendation. World Wide Web Consortium. URL: https://www.w3.org/TR/2025/REC-vc-data-model-2.0-20250515/ (дата звернення: 15.10.2026).

3. Decentralized Identifiers (DIDs) v1.0: Core architecture, data model, and representations. *W3C* Recommendation. World Wide Web Consortium. URL: https://www.w3.org/TR/2022/REC-did-core-20220719/ (дата звернення: 15.10.2026).

4. Hu V. C., Ferraiolo D., Kuhn R. et al. Guide to Attribute-Based Access Control (ABAC): Definition and Considerations. NIST Special Publication 800-162. Gaithersburg : NIST, 2014 (updated 2019). DOI: 10.6028/NIST.SP.800-162

5. Verifiable Credential Data Integrity 1.0. W3C Recommendation. World Wide Web Consortium. URL: https://www.w3.org/TR/2025/REC-vc-data-integrity-20250515/ (дата звернення: 16.10.2026).

УДК 004.93

Дмитро Чередниченко
здобувач другого (магістерського) рівня вищої освіти
спеціальності «Комп'ютерна інженерія»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євген Дарнапук
PhD, доцент,
в.о. завідувача кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ОГЛЯД ТЕХНОЛОГІЙ КОМП'ЮТЕРНОГО ЗОРУ В EDGE-AI СИСТЕМАХ

Інтеграція штучного інтелекту (ШІ) та Інтернету речей (IoT) вимагає суттєвої оптимізації процесів обробки візуальних даних. Класичний підхід до завдань комп'ютерного зору (Computer Vision) базується на хмарних обчисленнях (Cloud-AI), де кінцеві пристрої виконують лише функцію трансляції відеопотоку. Передача масивів неструктурованих даних на віддалені сервери має критичні недоліки: високі затримки мережі (latency), залежність від стабільності Інтернет-з'єднання, експоненційне зростання витрат на пропускну здатність каналів зв'язку та ризики витоку конфіденційної інформації [4].

Ефективним вирішенням цих проблем є перехід до концепції Edge-AI (периферійний штучний інтелект).

Технологія Edge-AI забезпечує виконання алгоритмів машинного навчання та математичну обробку зображень безпосередньо на кінцевих пристроях (смарт-камерах, мікроконтролерах, сенсорних вузлах) [5]. Впровадження таких систем гарантує повну автономність роботи пристроїв, мінімізує час реакції на події до мілісекунд та кардинально знижує навантаження на магістральну мережеву інфраструктуру.

Реалізація периферійних обчислень стала можливою завдяки швидкому розвитку апаратної бази та появі спеціалізованих методів стиснення моделей глибокого навчання. Сучасні мікроконтролери та одноплатні комп'ютери мають достатню обчислювальну потужність для локальної роботи із заздалегідь натренованими нейромережами без залучення зовнішніх серверних ресурсів [1]. Для розгортання Edge-AI рішень використовують різні класи пристроїв, що відрізняються архітектурою, обсягом пам'яті та обчислювальною продуктивністю (табл. 1).

Таблиця 1. – Апаратні платформи для систем комп'ютерного зору на базі Edge-AI

№ з/п	**Платформа**	**Практичне застосування у системах комп'ютерного зору**	**Технічні характеристики**
1	ESP32-CAM	Детекція облич, класифікація базових об'єктів (до 10–15 кадр/с)	Процесор 240 МГц, 4 Мбайт PSRAM, камера OV2640
2	OpenMV Cam H7	Кольоровий трекінг, читання штрих-кодів, базова класифікація	Процесор Cortex-M7 480 МГц, 1 Мбайт ОЗП, підтримка TensorFlow Lite
3	Sipeed Maix Bit	Апаратне розпізнавання об'єктів та облич у реальному часі (до 30 кадр/с)	2-ядерний процесор RISC-V 400 МГц, вбудований KPU (нейропроцесор), 8 Мбайт SRAM
4	Raspberry Pi 4	Локальний трекінг об'єктів, обробка відео 1080p	Процесор ARM Cortex-A72, до 8 Гбайт ОЗП, інтеграція з IoT-екосистемою

№ з/п	Платформа	Практичне застосування у системах комп'ютерного зору	Технічні характеристики
5	Google Coral Dev Board	Обробка складних моделей у реальному часі (понад 100 кадр/с)	Процесор NXP i.MX 8M, співпроцесор Edge TPU (4 TOPS)
6	NVIDIA Jetson Nano	Складна аналітика для кількох відеопотоків одночасно	Графічний процесор (128 ядер Maxwell), 4 Гбайт ОЗП, апаратне прискорення ШІ

Вибір платформи залежить від специфіки завдання. Наприклад, плати рівня NVIDIA Jetson або Google Coral використовують спеціалізовані тензорні співпроцесори (TPU), які апаратно прискорюють операції множення та накопичення (MAC) – базові математичні операції для згорткових нейронних мереж. Однак для побудови масових, енергоефективних та економічно вигідних кінцевих вузлів (end-nodes) доцільно використовувати мікроконтролерні платформи. Цей клас пристроїв інтегрує обчислювальні ядра, необхідні інтерфейси та підсистеми енергозбереження на одному кристалі.

Розширення апаратних можливостей мікроконтролерів за рахунок додаткової пам'яті PSRAM (Pseudo-Static RAM) є критично важливим фактором для виконання завдань комп'ютерного зору на периферії. Внутрішньої оперативної пам'яті (SRAM) базових чипів (зазвичай від 320 до 512 Кбайт) фізично недостатньо для зберігання навіть одного нестисненого відеокадру з роздільною здатністю VGA у кольоровому форматі RGB565. Використання зовнішніх мікросхем PSRAM об'ємом 4 Мбайт або 8 Мбайт, підключених через високошвидкісний інтерфейс QSPI, дозволяє системі динамічно виділяти необхідні фрейм-буфери (Frame Buffers) для захоплення відеопотоку та розміщувати тензори нейромереж безпосередньо в адресному просторі мікроконтролера.

Окрім того, інтеграція оптичних сенсорів реалізується через паралельний інтерфейс DVP (Digital Video Port) із підтримкою механізму прямого доступу до пам'яті (DMA). Такий підхід апаратно розвантажує ядро центрального процесора від попіксельного зчитування матриці. Процесор отримує можливість виконувати складні математичні розрахунки інференсу або підтримувати мережеве з'єднання паралельно з тим, як DMA-контролер заповнює буфер новим кадром.

Стандартом серед таких пристроїв став модуль ESP32-CAM (рис. 1), який поєднує двоядерний процесор ESP32, 4 Мбайт PSRAM, модулі Wi-Fi/Bluetooth та камеру стандарту OV2640.



Рисунок 1. – Модуль розробника ESP32-CAM з камерою OV2640

Процес розгортання моделей комп'ютерного зору на периферійних пристроях вимагає специфічного програмного стеку, відомого як TinyML (Tiny Machine Learning). Стандартний цикл розробки включає етап тренування повноцінної моделі на високопродуктивних хмарних серверах, її подальшу оптимізацію та конвертацію у мікроформат. Для роботи з візуальними даними на мікроконтролерах найвищу ефективність демонструють оптимізовані архітектури згорткових нейронних мереж (CNN), зокрема легковагові моделі сімейства MobileNet (V1/V2) та YOLO-Tiny [3]. Значним кроком у розвитку напрямку стала поява MLOps-платформ, які дозволяють автоматизувати процес збору даних, тренування та генерації оптимізованого C++ коду безпосередньо під конкретний мікроконтролер.

Апаратні обмеження ESP32-CAM компенсуються використанням фреймворку TensorFlow Lite for Microcontrollers. Цей інструмент оптимізує споживання оперативної пам'яті за рахунок статичного виділення буферів (Tensor Arena) та використання структури даних FlatBuffers, що дозволяє уникнути динамічного розподілу пам'яті під час виконання програми.

Процес локальної обробки візуальних даних (пайплайн) на пристрої ESP32-CAM складається з чіткої послідовності етапів:

1. Сенсор OV2640 генерує кадр, який через інтерфейс прямого доступу до пам'яті (DMA) передається у буфер PSRAM.

2. Зображення декодується з формату JPEG (або RGB565), обрізається та масштабується (resizing) до розміру вхідного тензора нейромережі (зазвичай 96×96 або 160×160 пікселів). Далі відбувається нормалізація значень кольорів матриці у діапазон [0, 1] або [-1, 1].

3. Навчена модель виконує прямий прохід (forward pass) матриці через усі шари нейромережі.

4. Система аналізує вихідний тензор, визначаючи координати об'єкта (bounding boxes) та ймовірність його приналежності до певного класу (confidence score) [2].

Аналіз кадру відбувається виключно локально (on-device), а через бездротову мережу на центральний сервер передається лише результат розпізнавання у вигляді текстового JSON-пакета або тригерного сигналу, розмір якого становить лічені байти.

Для досягнення максимальної автономності та енергоефективності вузлів на базі мікроконтролерів застосовується подійно-орієнтована архітектура (event-driven architecture). У штатному режимі мікроконтролер ESP32 перебуває у стані глибокого сну (Deep Sleep), відключивши ядро, периферію та камеру, споживаючи при цьому струм на рівні кількох мікроампер. Активація системи відбувається миттєво за апаратним перериванням від зовнішнього інфрачервоного датчика руху (PIR-сенсора). Після пробудження модуль захоплює кадр, виконує аналіз, відправляє результати через легковаговий протокол (наприклад, MQTT) і миттєво повертається у режим сну.

Ключові експлуатаційні переваги такого підходу у системах Edge-AI:

– Забезпечення високого рівня конфіденційності. Відсутність постійної трансляції відеопотоку унеможливлює перехоплення приватної інформації (зокрема, біометричних даних) на проміжних вузлах мережі;

– Економія пропускної здатності. Передача кілобайтів телеметрії замість гігабайтів відеоданих дозволяє використовувати для зв'язку енергоефективні мережі стандарту LoRaWAN або NB-IoT;

– Висока масштабованість. Низька вартість компонентів та енергетична автономність дозволяють розгортати тисячі розумних сенсорів у межах інфраструктури «розумного міста» (Smart City) або промислового Інтернету речей (IIoT).

Головним технологічним викликом при роботі з Edge-AI системами залишається проблема втрати точності моделей (Accuracy Drop) під час їхнього стиснення. Для перенесення нейромережі на ESP32-CAM обов'язковою є процедура квантизації (Post-Training Quantization) – зниження розрядності вагових коефіцієнтів мережі з 32-бітного формату з плаваючою комою (FP32) до 8-бітного цілого числа (INT8). Це дозволяє зменшити обсяг пам'яті, необхідний для зберігання моделі, у 4 рази та прискорити обчислення, проте потребує

калібрування на репрезентативному наборі даних для збереження якості розпізнавання [1; 5]. Крім того, на етапі тренування активно застосовується алгоритм прунінгу (Pruning) – примусове видалення найменш значущих нейронних зв'язків.

Список використаних джерел

1. Мельник О. В., Коваленко І. М. Інтеграція систем комп'ютерного зору в інфраструктуру Інтернету речей на базі мікроконтролерів. *Сучасні інформаційні системи.* 2023. Том 7, № 2. С. 45–52.
2. Ткаченко В. П., Сидоренко О. О. Аналіз ефективності периферійних обчислень (Edge Computing) для локальної обробки відеоданих. *Вісник Національного технічного університету «ХПІ». Серія : Інформатика та моделювання.* 2024. № 1. С. 112–118.
3. Ettalibi A., Elouadi A., Mansour A. AI and Computer Vision-based Real-time Quality Control: A Review of Industrial Applications. *Procedia Computer Science.* 2024. Vol. 231. P. 212–220. DOI: 10.1016/j.procs.2023.12.195
5. Iftikhar S., Gill S. S., et al. AI-based fog and edge computing: A systematic review, taxonomy and future directions. *Internet of Things.* 2023. Vol. 21. P. 100674. DOI: 10.1016/j.iot.2022.100674
6. Murshed M. G. S., Murphy C., Hou D. Machine learning at the network edge: A survey. *ACM Computing Surveys.* 2021. Vol. 54, No. 8. P. 1–37. DOI: 10.1145/3469029

УДК 004.89:629.7.05

Гліб Чернигін
аспірант спеціальності «Комп'ютерна інженерія»
Ірина Журавська
доктор технічних наук, професор,
професор кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

СИСТЕМА АВТОНОМНОГО ПЛАНУВАННЯ І ДИНАМІЧНОГО КОРИГУВАННЯ ЗАВДАНЬ БПЛА НА БАЗІ АГЕНТНОГО ШТУЧНОГО ІНТЕЛЕКТУ

У сучасному суспільстві стрімкий розвиток мобільних кіберфізичних систем (МКФС), зокрема безпілотних літальних апаратів (БпЛА), формує нові парадигми, як-от Low-Altitude Economy [1]. Застосування таких систем постійно розширюється, охоплюючи сільське господарство, логістику, пошуково-рятувальні операції, моніторинг екологічних катастроф, гарантування безпеки тощо. Разом зі стрімким зростанням кількості застосувань зростає й їхня функціональна складність. У світовій робототехніці відбувається парадигмальний перехід від прямого керування до концепції делегованої автономії, за якої оператор встановлює лише високорівневі абстрактні цілі [2]. Проте найпоширеніші сучасні станції керування розглядають планування завдань здебільшого як інженерно-геометричну задачу на плоских 2D- або 3D-мапах. Цей підхід вимагає від оператора топографічної обізнаності, ручного розрахунку маршруту польоту та постійного аналізу процесу виконання місії. Як наслідок, на оператора створюється високе когнітивне навантаження, яке є особливо критичним під час тривалих місій у динамічних умовах.

Відсутність гнучкості стає критичною проблемою в умовах стохастичного середовища, що наштовхує на потребу в переході наземної станції керування БпЛА від класичного «командного пункту» до повноцінного інтелектуального когнітивного вузла на базі агентних систем штучного інтелекту (ШІ), ядром яких є великі мовні моделі (англ. Large Language Model, LLM) [1; 2].

Тому актуальною на теперішній час є ідея системи автономного планування та динамічного коригування завдань МКФС на базі агентного ШІ, яка дозволить суттєво підвищити рівень операційної автономії цих систем і мінімізувати когнітивне навантаження на

оператора. Архітектура когнітивного вузла керування має бути побудована не навколо парадигми «клієнт-сервер», а як автономна система (рис. 1). В такій системі LLM інтегрована як механізм логічного міркування (англ. Reasoning Engine).

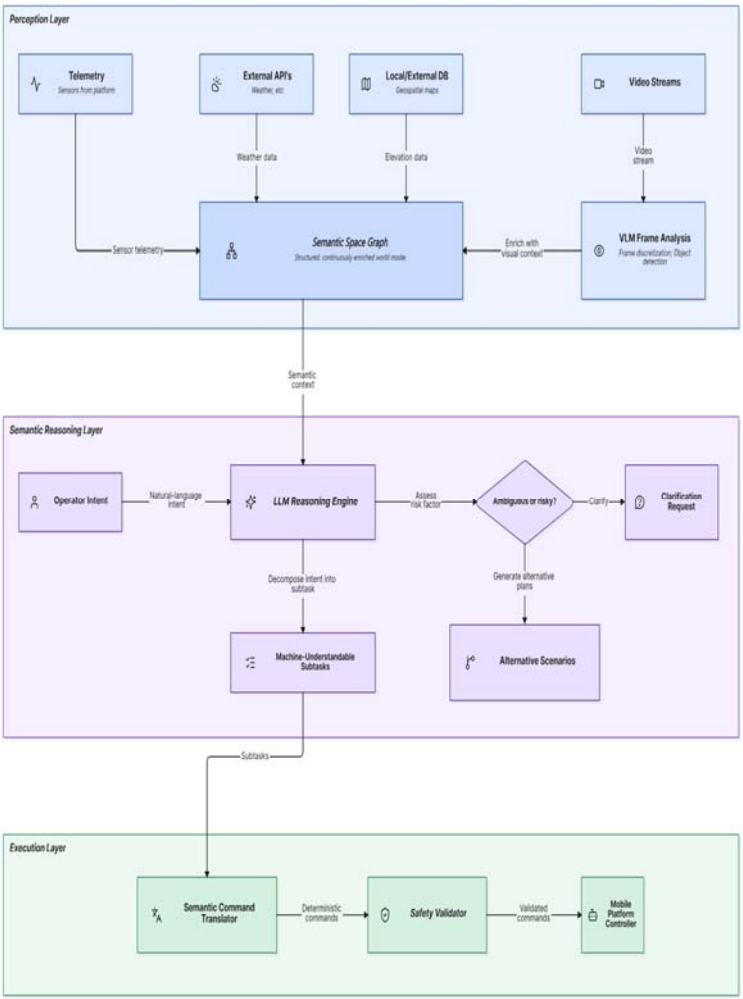


Рисунок 1. – Архітектура когнітивного вузла

Архітектура такого когнітивного вузла керування може складатися з трьох ключових рівнів (рис. 1):

1) Рівень сприйняття (англ. Perception Layer) – відповідає за безперервну агрегацію та структурування мультимодальних потоків даних. Збирає телеметрію, завантажує прогнози погоди з зовнішніх API, отримує топографічні висотні мапи з баз даних GIS та обробляє візуальні відеопотоки [3]. Дискретизація відеокадрів дозволяє системі розпізнавати незадокументовані на картах перешкоди (будівельні крани, тимчасові лінії електропередачі) та збагачувати семантичний граф простору.

2) Рівень семантичного міркування (англ. Semantic Reasoning Layer) – інтелектуальне ядро, яке здійснює семантичну декомпозицію завдань на зрозумілі машині підзадачі за допомогою ланцюга міркувань (Chain-of-Thought). При неоднозначних або потенційно небезпечних намірах оператора ШІ проактивно запитує уточнення оператора або генерує альтернативні сценарії виконання [4].

3) Рівень виконання (англ. Execution Layer) – шар абстракції, де семантичні рішення перетворюються на детерміновані команди для контролерів мобільних платформ, відокремлюючи від оператора складність низькорівневих апаратних протоколів і переводячи взаємодію на природний рівень людської мови.

Для подолання фрагментації апаратних протоколів у базову архітектуру впроваджується стандарт Model Context Protocol (MCP). Спеціалізовані MCP-сервери виступають надійними мостами між агентним ШІ, хмарними сервісами, локальними базами даних та іншими інструментами. Трансляція високорівневих намірів оператора здійснюється за допомогою методу Agentic RAG: від семантичного парсингу та пошуку в базі геоданих до формування готового покрокового рішення на основі отриманого контексту.

Для прийняття безпечних рішень під час виконання завдань потрібен компонент валідації згенерованих команд, який буде стійкий до галюцинацій і не дозволить виконати потенційно небезпечну команду без підтвердження оператора. Перспективними технологіями в цьому напрямку є моделі класу System One [5], а також класичні математичні класифікатори.

Безперервне перепланування завдань пропонується реалізовувати на базі гібриду Edge та Cloud Computing із використанням квантованих локальних мовних моделей (SLM) [6] і потужних мовних моделей у хмарі для делегування складних завдань.

Алгоритмічно запропонована система спирається на адаптований для мовних моделей цикл Бойда [7], який ґрунтується на модифікованій версії класичної концепції прийняття рішень «Спостерігай – Орієнтуйся – Вирішуй – Дій» або англ. OODA loop: Observe, Orient, Decide, Act (рис. 2).

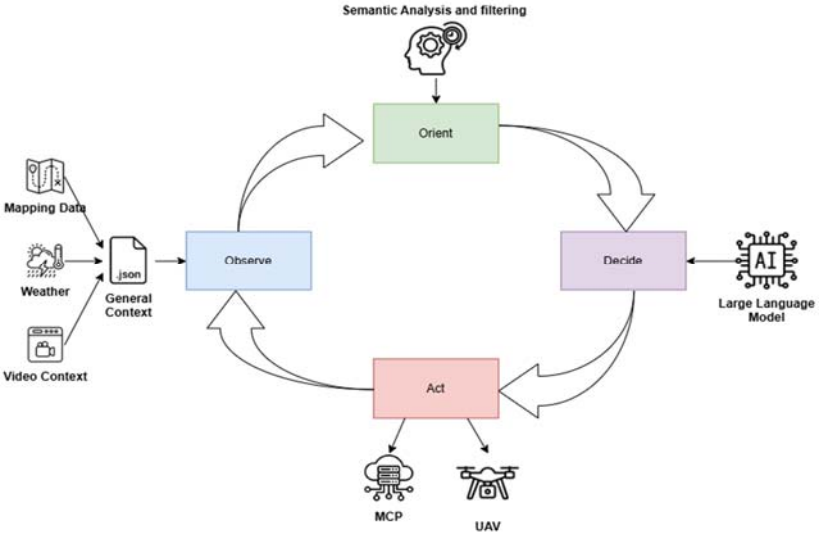


Рисунок 2. – Адаптований цикл OODA

На етапі «Сприйняття» (Observe) формується єдиний стислий JSON-контекст. Етап «Орієнтація» (Orient) діє як інтелектуальний фільтр, виявляючи критичні семантичні аномалії та запобігаючи перевантаженню LLM дрібними змінами. На етапі «Міркування» (Decide) SLM генерує альтернативні варіанти дій за критеріями безпеки та енергоефективності, а на етапі «Дія» (Act) обраний варіант автоматично трансформується у виклик MCP-інструменту або команду апарату. Для гарантування мінімальної затримки в першу чергу ефективним є використання локальної квантованої моделі, яка за потреби звертається до потужніших моделей у хмарі.

Запропонована система здатна проактивно генерувати оптимальний безпечний маршрут виконання місії, але оскільки впровадження недетермінованих LLM-систем несе ризики галюцинацій, критично важливою складовою є контур верифікації. Задача контуру – перевірити згенеровану агентним ШІ команду на відповідність

фізичним обмеженням апарату та середовища. Лише після такого підтвердження команда передається на виконання або на уточнення до оператора.

Розглянутий підхід до побудови архітектури агентної наземної станції керування БпЛА дозволить здійснити якісний стрибок до найвищих рівнів інтелектуальної автономії (7-10 рівнів за шкалою ALFUS) [8], мінімізує потребу в постійному ручному контролі та суттєво поліпшить комплексне застосування мобільних кіберфізичних систем.

Список використаних джерел

1. Tian Y., Lin F, Li Yi., et al. UAVs meet LLMs: Overviews and perspectives toward agentic low-altitude mobility. *Information Fusion*. 2025. Vol. 122, Art. ID: 103158. DOI: 10.1016/j.inffus.2025.103158
2. Chen Yi., Que X., Zhang J., et al. When Large Language Models meet UAV projects: An empirical study from developers' perspective. arXiv:2509.12795 [cs.SE]. 2025. DOI: 10.48550/arXiv.2509.12795
3. Kassis T., Agarwal V., He Y., Patel D., Brueckner A. M. Scientific Agent Skills: A library of procedural knowledge for research agents. arXiv:2609.00065. 2026. DOI: 10.48550/arXiv.2609.00065
4. Худов Г. В., Хижняк І. А., Ярмолюк В. М., Марченко В. П. Методика обробки зображень з бортових систем спостереження в інтересах подальшої класифікації об'єктів інтересу. *Системи обробки інформації*. 2022. № 2 (169). С. 87–95. DOI: 10.30748/soi.2022.169.10
5. What is a System One (System 1) model? *System One Models.* Publ. Oct. 02, 2026. URL: https://systemonemodels.org/guides/what-is-a-system-one-model/ (дата звернення: 04.10.2026).
6. Yang J., Wu Q., Feng Z., et al. Quality-of-Service aware LLM routing for edge computing with multiple experts. arXiv:2508.00234 [cs.NI]. 2025. DOI: 10.48550/arXiv.2508.00234
7. Зосим М. Петля Бойда (ODA loop). Опубл. 02.03.2023. URL: https://www.maxzosim.com/oda-loop/ (дата звернення: 06.10.2026).
8. Huang H.-M., Messina E. R. Autonomy Levels For Unmanned Systems (ALFUS) framework. Volume II: Framework models initial version. *NIST Special Publication*. 2007. Report Number: 1011-II-1.0. DOI: 10.6028/NIST.SP.1011-II-1.0

UDC 004.38:004.052.2:519.673

Gennady Chuiko
D. Sc., professor
Petro Mohyla Black Sea National University
Mykolaiv, Ukraine
Yevhen Darnapuk
PhD, assoc. professor
Petro Mohyla Black Sea National University
Mykolaiv, Ukraine
Polina Kravchenko
Master's Student
Petro Mohyla Black Sea National University
Mykolaiv, Ukraine

DIAGNOSTICS AND CERTIFICATION OF PAULI QUANTUM CHANNELS: ENGINEERING ASPECTS

A quantum channel can be viewed as a physical device, similar to a pipe, through which quantum information is transmitted. A qubit enters the channel in each quantum state, interacts with its environment during transmission, and appears at the output in a generally altered state. From an engineering perspective, the channel may therefore be regarded as a device whose condition and performance can be diagnosed and certified.

In an idealized world, the output state would be identical to the input state. In the real world, however, interactions with the environment introduce quantum noise. As a result, the transmitted qubit may become partially or completely degraded, losing information, coherence, or quantum correlations.

From an engineering perspective, the quantum channel can therefore be treated as a device whose condition must be diagnosed. The purpose of channel diagnostics is to decide how strongly the transmitted qubit has been affected by noise and to describe the channel using a standardized certificate.

A Pauli channel is one of the most important models of single-qubit noise. It assumes that the transmitted qubit stays unchanged with probability p_0 while Pauli errors X, Y, Z occur with probabilities p_x, p_y, p_z respectively. Let ρ denote the density matrix of the input state. The output state is then obtained according to

$$\Phi(\rho) = p_0\rho + p_x X\rho X + p_y Y\rho Y + p_z Z\rho Z \qquad (1)$$

where X, Y, Z are the Pauli matrices, and $p_0 + p_x + p_y + p_z = 1;\ p_i \geq 0$, which specify the likelihood of each possible error acting on the transmitted qubit.

Although a Pauli channel is naturally specified by above four the probabilities ($p = \left(p_0, p_x, p_y, p_z\right)^T$) it is often more convenient to describe the same channel by the Bloch-contraction vector $\lambda = (\lambda_x, \lambda_y, \lambda_z)^T$. The two representations are related by the affine transformation:

$$\lambda = \begin{pmatrix} 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} p \tag{2}$$

Thus, the probability vector and the contraction vector provide two equivalent descriptions of the same Pauli channel. The probability representation is convenient for specifying noise processes, while the contraction representation is convenient for geometric analysis. Besides their geometric representation, Pauli channels can also be classified according to their operational properties. Operational classes characterize how effectively a quantum channel can preserve and transmit quantum information. Depending on the strength and nature of the noise, a channel may destroy quantum correlations, partially preserve them, or keep enough information for quantum communication tasks. Three important classes are commonly used in quantum information theory:

1) Entanglement-Breaking (EB) channels;
2) Antidegradable (AD) channels;
3) Degradable (D) channels.

An EB channel destroys all quantum entanglement between the transmitted qubit and any external quantum system. Such channels correspond to the strongest levels of quantum noise. AD channels are affected by noise so strongly that the environment receives at least as much information as the channel output. D channels preserve quantum information sufficiently well so that the information available to the environment can be reconstructed from the channel output. The relationships between the three classes are summarized as follows:

- every EB channel is also AD;
- not every AD channel is EB;
- D and AD are not complementary independent classes;
- some channels mat belongs to both D and AD classes;
- some channels belong to neither class.

In engineering practice, the condition and performance of a technical device are often summarized in a standardized certificate having its key characteristics and diagnostic results. A similar approach can be applied to quantum channels. A Pauli quantum channel may be viewed as a communication device that affects transmitted quantum information. Different channels introduce different levels and types of noise and therefore require systematic characterization.

In this report, a channel certificate is a standardized diagnostic record that summarizes the most important properties of a Pauli channel. The certificate combines information obtained from probability, geometric, operational, and informational analyses into a single report the certification workflow used in this report is:

$Pauli\ Channel -> Diagnostics -> Classification -> Channel\ Certificate$

Supposer, that a Pauli channel with probability vector

$$p = (0.55, 0.12, 0.15, 0.18)^T \tag{4}$$

is given for engineering inspection. The purpose of the inspection is to decide the geometric, operational, and informational characteristics of the channel and to generate a standardized channel certificate. Figure 1 shows column graphs of components of probabilities and Bloch-contractions vectors:

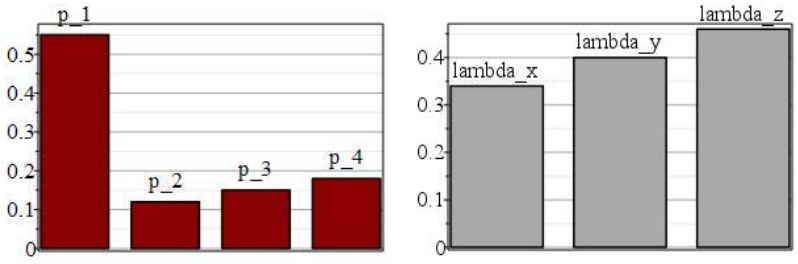


Figure 1. – Column graphs for components of probability vector (left panel) and Bloch-contraction on (right panel)

Since fisty component is the largest probability, the channel belongs to the Identity (I) sector. The weakest contraction coefficient ($\lambda_x = 0.34$) defines the axis with largest contraction loss. The root-mean-square equatorial retention coherence is:

$$CRI_Z = \sqrt{\frac{\lambda_x^2 + \lambda_y^2}{2}} \quad (5)$$

The binary entropy of the probability vector is

$$H = -\sum_{i=1}^{4} p_i log_2 p_i \quad (6)$$

The entropy allows to calculate hashing expressions, particularly the hashing bound (R_{hash}). The hashing expression provides an approximate information-theoretic sign of the channel's ability to support quantum communication. Larger values correspond to channels that preserve quantum information more effectively, while negative values show strong information loss.

The final stage of the certification procedure is the compilation of all previously calculated characteristics into a standardized channel certificate. The results obtained during geometric and operational diagnostics, as well as descriptor evaluations, are summarized. Table 1 presents the channel certificate for the example picked out above.

Table 1. – Example of a Pauli channel certificate

Field	**Value**
Probability Vector	(0.55, 0.12, 0.15, 0.18)
Bloch-Contraction Vector	(0.34, 0.40, 0.46)
EB Status	No
AD Status	Yes
D Status	No
Operational Status	AD only
Orientation	Preserving
Sector	Identity
q_min	0.34
l_max	0.66
CRI_Z	0.371214
I_hash	-0.697293
R_hash	0

The generated certificate provides a compact description of the channel and combines its probability, geometric, operational, and informational characteristics in a single record.

From the certificate we conclude that the channel:

– is a physically valid Pauli channel;
– is not entanglement breaking;
– is antidegradable but not degradable;
– belongs to the Identity sector;
– preserves the orientation of Bloch space;
– exhibits moderate contraction and limited capability for quantum information transmission.

In practical applications, all certification steps can be combined into a single software procedure. Nevertheless, we reckon that understanding the physical meaning of each certificate field is more important than the automation itself.

Bibliography

1. Fujiwara A., Algoet P. One-to-one parametrization of quantum channels. *Physical Review A.* 1999. Vol. 59, No. 5. P. 3290–3294. DOI: 10.1103/physreva.59.3290

2. Beth Ruskai M., Szarek S., Werner E. An analysis of completely-positive trace-preserving maps on M2. *Linear Algebra and its Applications*. 2002. Vol. 347, No. 1-3. P. 159–187. DOI: 10.1016/s0024-3795(01)00547-x

3. Cubitt T. S., Ruskai M. B., Smith G. The structure of degradable quantum channels. *Journal of Mathematical Physics*. 2008. Vol. 49, No. 10. P. 102104. DOI: 10.1063/1.2953685

4. Chuiko G. P., Darnapuk Y. S., Kravchenko P. K. Geometry, symmetry, and topology of orientation-reversing Pauli channels. *Physics Letters A*. 2026. P. 132032. DOI: 10.1016/j.physleta.2026.132032

УДК 004.75:616-073.7

Віктор Шевченко
здобувач другого (магістерського) рівня вищої освіти
спеціальності «Автоматизація,
комп'ютерно-інтегровані технології та робототехніка»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євген Дарнапук
PhD, доцент,
в.о. завідувача кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ПРАКТИЧНА ОЦІНКА МАСШТАБОВАНОСТІ ТА СТІЙКОСТІ СИСТЕМИ РОЗПОДІЛЕНОГО ОБМІНУ DICOM-ЗОБРАЖЕННЯМИ

Передавання медичних зображень між діагностичними пристроями, локальними робочими станціями та архівними серверами потребує узгодженого формату даних і надійного мережевого обміну. Стандарт DICOM визначає структуру зображень та пов'язаних із ними атрибутів [1], а PACS забезпечує їх приймання, зберігання й подальший пошук. У невеликих медичних підрозділах і навчально-дослідних лабораторіях повноцінна серверна інфраструктура не завжди є виправданою. Одним із практичних варіантів стає розподіл функцій між компактним периферійним вузлом і центральним архівом [2].

У такій архітектурі питання масштабованості та стійкості не можна оцінювати лише за фактом успішного передавання окремого файлу. Масштабованість пов'язана зі здатністю системи працювати зі збільшенням кількості об'єктів, запитів або вузлів, а стійкість – зі збереженням передбачуваної роботи за повторних операцій, мережевих затримок і можливих відмов. Для малоресурсного вузла обмеження можуть виникати не лише на рівні каналу зв'язку, а й через завантаження процесора, витрати оперативної пам'яті та оброблення помилок клієнтською програмою.

Метою даного дослідження є практична перевірка вихідної працездатності розподіленої системи обміну DICOM-зображеннями та визначення показників, за якими можна обґрунтовано оцінювати її подальше масштабування і стійкість. За основу взято прототип із

периферійним вузлом Raspberry Pi 4 та PACS-сервером Orthanc, розгорнутим на персональному комп'ютері. З'єднання між ними організовано в локальній мережі Wi-Fi. Система використовує клієнтський модуль Python із бібліотекою requests та HTTP-інтерфейс Orthanc, що дає можливість виконувати автоматизовані операції з DICOM-файлами [3].

Перевірка була побудована як послідовність дій із фіксацією відповіді сервера. Спочатку з Raspberry Pi 4 надсилали запит GET до ресурсу /system. Отриманий код HTTP 200 підтвердив доступність Orthanc, а успішне проходження автентифікації – коректність облікових даних. Далі клієнтський модуль надсилав DICOM-об'єкт методом POST до ресурсу /instances. У відповіді сервера отримано HTTP 200 та повідомлення Status: Success; прийнятий об'єкт було знайдено через Orthanc Explorer. Це підтвердило завершеність маршруту від локального файлу до серверного архіву.

До набору випробувальних даних включено файли CT_small.dcm (39 206 байт), MR_small.dcm (9 830 байт) та SC_rgb_small_odd.dcm (1 444 байти) з тестового комплекту pydicom. Вони мають різну структуру та розмір, тому дають можливість перевіряти роботу з кількома типами DICOM-об'єктів. Файл SC належить до класу Secondary Capture і використовується як технічний тестовий приклад, а не як повноцінна клінічна рентгенограма. Послідовне надсилання кількох файлів обрано замість паралельного: воно спрощує зіставлення результатів і не створює додаткової конкуренції за ресурси Raspberry Pi 4.

Критерії та фактичні результати функціональної перевірки зведено в табл. 1. Окремо вказано, що саме перевірялося безпосередньо, а які характеристики потребують наступного експериментального етапу.

Таблиця 1 – Узагальнення перевірки працездатності й меж оцінювання системи

Напрям перевірки	**Спосіб оцінювання**	**Встановлений результат**
Доступність вузла	GET /system, автентифікація	HTTP 200; сервер доступний з Raspberry Pi 4
Передавання й архівування	POST /instances, перегляд в Orthanc Explorer	HTTP 200, Status: Success; DICOM-об'єкт зареєстровано

Напрям перевірки	Спосіб оцінювання	Встановлений результат
Послідовний обмін	Надсилання тестових CT, MR та SC-файлів	У стенді зафіксовано успішні операції імпорту
Масштабування і відмови	Одночасні клієнти, зростання черги, розрив Wi-Fi	Кількісна перевірка цих режимів не проводилася

Результати, наведені у табл. 1, засвідчують функціональну стійкість прототипу в перевіреному штатному сценарії: клієнтський вузол установлює зв'язок, передає дані та отримує підтвердження від центрального сервера. Для медичних зображень істотним є не тільки код відповіді HTTP, але й те, що DICOM-об'єкт стає доступним у PACS-архіві з метаданими дослідження. Сам по собі успішний імпорт не доводить незмінності всіх піксельних значень; для цього потрібне окреме побітове зіставлення вихідних і відновлених даних.

Під час інтерпретації результатів було розділено три рівні оцінювання. Перший – працездатність одиничного маршруту Raspberry Pi 4 → Orthanc; вона підтверджена експериментально. Другий – повторюваність операцій із кількома тестовими об'єктами, яку можна досліджувати послідовними запусками клієнтського модуля. Третій – масштабованість під одночасним навантаженням, для якої потрібен окремий експеримент із кількома незалежними клієнтами, визначенням пропускної здатності та часу очікування. Дані першого й другого рівнів не слід автоматично переносити на третій.

Для подальшої кількісної оцінки доцільно задавати контрольоване навантаження серіями однакових за обсягом файлів, окремо змінюючи кількість клієнтів і кількість DICOM-об'єктів. Вимірюваними величинами є тривалість імпорту, середня та максимальна затримка, частка завершених без помилок запитів, фактичний обсяг переданих даних, використання CPU та оперативної пам'яті периферійного вузла. За зростання навантаження можна визначити, на якому етапі з'являються черги або помилки та чи обмежувальним чинником стає Wi-Fi, Raspberry Pi 4 або Orthanc.

Окремий сценарій стійкості передбачає короткочасне роз'єднання Wi-Fi або тимчасову недоступність центрального сервера. У цьому випадку необхідно перевіряти те, чи зберігається локальний файл, чи повторне надсилання не призводить до небажаного дублювання дослідження та чи фіксується причина помилки в журналі клієнтського модуля. Такий тест не виконувався в межах наявної базової перевірки,

тому твердження про автоматичне відновлення після відмови було б передчасним. Для промислового використання також потрібне шифрування трафіку: базова HTTP-автентифікація без TLS сама собою не гарантує конфіденційності медичних даних. Оцінка масштабованості пов'язана і з безвтратним стисканням. Зменшення обсягу передаваних піксельних даних потенційно скорочує мережеве навантаження, але створює додаткові обчислення на периферійному вузлі. Тому CBPC, CBPC2 та ZIP коректно порівнювати за коефіцієнтом стискання, тривалістю кодування і декодування та тотожністю відновлених даних. У наявному стенді підтверджено передусім роботу каналу обміну DICOM; наведені міркування щодо стискання визначають необхідні умови окремих порівняльних випробувань, а не замінюють їх числові результати.

За фіксації пропускної здатності слід ураховувати, що один запит POST містить не тільки піксельні дані, а й службову частину DICOM-файлу. Для невеликих тестових об'єктів відносні витрати на встановлення з'єднання й оброблення запиту можуть бути істотними. Тому порівняння лише часу надсилання файлів різного розміру недостатнє. Потрібно окремо визначати час відповіді сервера, обсяг даних і кількість завершених операцій за однаковий проміжок часу. Це дає можливість відокремити обмеження, пов'язані з розміром набору, від обмежень клієнтського та серверного програмного забезпечення.

При переході від одного Raspberry Pi 4 до кількох периферійних вузлів змінюється не тільки інтенсивність мережевого трафіку. На центральному Orthanc одночасно виконуються перевірка запитів, запис об'єктів та оновлення індексів архіву. У такому режимі необхідно контролювати затримки в черзі імпорту, заповнення сховища й кількість повторних спроб. Базова архітектура допускає підключення додаткових клієнтів через той самий інтерфейс REST API, проте сам факт такої можливості не є свідченням лінійного зростання продуктивності системи.

Відтворюваність вимірювань залежить і від організації самого випробування. Доцільно фіксувати версії програм, характеристики Raspberry Pi 4, умови бездротового з'єднання, початковий стан архіву та однаковий набір DICOM-об'єктів для кожної серії. Особливої уваги потребує повторне надсилання об'єкта з тим самим ідентифікатором SOP Instance UID: сервер може розпізнати вже наявний файл, через що повторний імпорт і завантаження нового об'єкта не завжди еквівалентні як навантажувальні операції. Тому для дослідження продуктивності необхідно чітко відокремлювати повторну перевірку каналу від формування нового навантаження на архів.

Практичний результат роботи полягає в отриманні відтворюваної конфігурації з Raspberry Pi 4 та Orthanc, на якій перевірено автентифікований доступ до PACS, приймання тестових DICOM-

файлів і їх архівування. Такий стенд може використовуватися для поступового розширення експериментів без перебудови базової архітектури. Водночас проведені тести підтверджують стійкість системи лише у штатних умовах невеликого навантаження. Повноцінна оцінка масштабованості й відмовостійкості вимагає навантажувальних запусків, ін'єкції мережевих збоїв і зіставлення виміряних характеристик із визначеними порогами прийнятності.

Список використаних джерел

1. DICOM Standards Committee. DICOM *PS3.1: Introduction and Overview*. URL: https://dicom.nema.org/medical/dicom /current/ output/html/part01.html (дата звернення: 19.10.2026).

2. Дарнапук Є. С., Шевченко В. В. Розподілена система обміну DICOM-зображеннями з безвтратним стисканням з використанням одноплатних комп'ютерів. *Могилянські читання – 2025* : тези доп. XXVIII Всеукр. наук.-практ. конф., Миколаїв, 10–14 листоп. 2025 р. Миколаїв : ЧНУ ім. Петра Могили, 2025. С. 117–121.

3. Jodogne S., Bernard C., Devillers M., Lenaerts E., Coucke P. Orthanc – A Lightweight, RESTful DICOM Server for Healthcare and Medical Research. *2013 IEEE 10th International Symposium on Biomedical Imaging.* 2013. P. 190–193. DOI: 10.1109/ISBI.2013.6556444

УДК 004.75:004.896:004.94

Костянтин Шеремет
аспірант спеціальності
«Автоматизація, комп'ютерно-інтегровані технології
та робототехніка»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Сергій Пузирьов
кандидат фіз.-мат. наук, доцент,
доцент кафедри комп'ютерної інженерії
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ІНТЕРАКТИВНА МАПА СТАНУ MESH-МЕРЕЖІ РУХОМИХ АВТОНОМНИХ МОДУЛІВ: КОНЦЕПЦІЯ НА ОСНОВІ ESP32 ТА LORA MESH

Функціонування груп рухомих автономних апаратів в умовах щільної міської забудови, складного рельєфу, аварійних ситуацій або порушень роботи телекомунікаційної інфраструктури супроводжується постійними змінами умов поширення радіосигналу та нестабільністю каналів зв'язку. Подібні виклики виникають під час координації мобільних сенсорних платформ, сервісних роботів і засобів міської кур'єрської логістики, коли покладатися виключно на покриття стільникових мереж (LTE/4G) неможливо або економічно недоцільно. За відсутності прямого каналу зв'язку з кожним окремим модулем ускладнюється диспетчерський контроль їхнього місцезнаходження, оцінка технічного стану апаратів і збір даних із бортових сенсорів.

Для організації автономного обміну даними можуть використовуватися децентралізовані мережі класу MANET (*Mobile Ad-hoc Networks*) на базі енергоефективної технології LoRa (*Long Range*) [1, 2]. Одним з варіантів реалізації такого підходу є багатострибкова топологія LoRa Mesh у якій мобільні модулі на базі мікроконтролерів ESP32 із відповідними радіотрансиверами LoRa можуть передавати пакети через сусідні вузли до базового шлюзу. На відміну від класичної зіркової архітектури LoRaWAN, така організація мережі дає змогу передавати дані через проміжні вузли без використання стільникової інфраструктури [3]. У фізичних радіомережах якість

таких з'єднань оцінюється за рівнем прийнятого сигналу (RSSI) та співвідношенням сигнал/шум (SNR) [2], а для роботи в умовах частих розривів зв'язку застосовуються принципи затримко-стійких мереж – DTN (*Delay-Tolerant Networking*), які базуються на буферизації та відкладеному пересиланні даних [4].

Для ефективного диспетчерського контролю таких груп необхідна розробка концепту хмарної геоінформаційної системи (Web-GIS), здатної комплексно візуалізувати стан mesh-мережі: поточне чи останнє відоме положення рухомих модулів, активні маршрути ретрансляції та просторовий розподіл зібраних сенсорних даних.

Концептуальна архітектура системи моніторингу. Запропонований концепт хмарного рішення (Cloud Concept) складається з трьох логічних рівнів:

1) Бортовий рівень (мобільні вузли): апаратні модулі на базі мікроконтролера ESP32 та трансивера LoRa (наприклад, SX1276/SX1262) [3]. Вузол періодично формує компактні пакети телеметрії, що містять географічні координати, внутрішній стан апарата (рівень заряду акумулятора, службовий статус) та поточні показники бортових датчиків. Оскільки технологія LoRa орієнтована на вузькосмугове передавання невеликих пакетів даних, обмін телеметрією здійснюється короткими повідомленнями, а у випадку тимчасової відсутності маршруту до шлюзу сформовані записи зберігаються у локальній пам'яті модуля.

2) Рівень шлюзу та серверної агрегації: базовий шлюз (Gateway) приймає пакети від досяжних вузлів мережі та пересилає їх до серверної частини [5]. Сервер оновлює відомий стан кожного мобільного агенту, фіксує час останнього сеансу зв'язку та обробляє пакети, що надійшли із затримкою після відновлення маршрутів.

3) Клієнтський рівень геовізуалізації (Web-GIS): інтерактивна веб-мапа на базі картографічних підкладок Google Maps та бібліотеки Leaflet/Folium, яка відображає динаміку переміщення вузлів, граф наявних радіоз'єднань та точковий шар отриманих вимірювань.

Узагальнену структурну схему взаємодії апаратних та програмних компонентів системи моніторингу наведено на рис. 1.

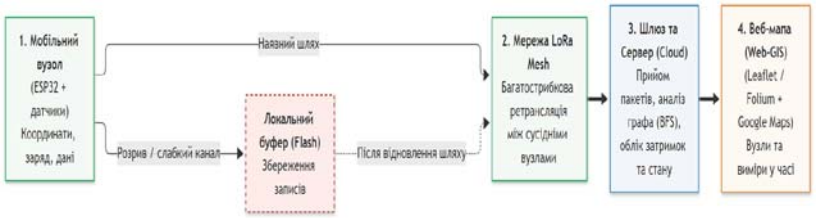


Рисунок 1. – Концептуальна схема поєднання технологій LoRa Mesh, локальної буферизації та Web-GIS візуалізації в системі моніторингу рухомих вузлів

Для апробації логіки серверної візуалізації розроблено дискретну імітаційну модель групи з п’яти вузлів у середовищі Google Colab мовою Python. Моделювання охоплює 12 часових тактів із кроком одна хвилина, а часова динаміка мережі відображається за допомогою плагіна TimestampedGeoJson.

Оскільки робота присвячена програмному моделюванню концепту візуалізації, у моделі прийнято низку спрощень:

– Геометрична умова зв’язності: реальна дальність зв’язку LoRa залежить від коефіцієнта розширення спектра, потужності передавача, параметрів антен, рельєфу та перешкод. В імітаційній моделі фізичне загасання сигналу та показники RSSI не розраховуються; натомість уведено умовне модельне припущення про граничну відстань зв’язку R_{max} = 210 м. Якщо геодезична відстань d_{ij} між вузлами *i* та *j* (за формулою гаверсинусів) не перевищує R_{max}, між ними формується ребро графа, яке на мапі забарвлюється зеленим (при d_{ij} менше ніж 0,65 R_{max}), або помаранчевим кольором (при d_{ij} більше ніж 0,65 R_{max}).

– Аналіз досяжності шлюзу: для перевірки наявності одно- або багатострибкового шляху від кожного модуля до шлюзу ESP32-GW на кожному такті використовується алгоритм пошуку в ширину (BFS) на поточному графі зв’язності. При цьому BFS виступає централізованим обчислювальним методом моделі для оцінки топології, а не мережевим протоколом маршрутизації чи протоколом DTN.

– Умовні сенсорні показники: для збереження універсальності моделі вимірювання бортових датчиків задано в абстрактних умовних одиницях (ум. од.) із розподілом за трьома категоріями для колірної індикації на мапі: базовий рівень (до 600 ум. од., зелений), підвищений

рівень (600-799 ум. од., помаранчевий) та умовна аномалія (від 800 ум. од., червоний).

– Розмежування фактичного та серверного станів: модель окремо відстежує фактичні параметри рухомого модуля та останні дані, відомі серверній частині. Якщо на поточному такті шлях до шлюзу відсутній, згенеровані вузлом записи зберігаються у локальному буфері (flash_buffer) і передаються до серверної колекції лише після повторної появи доступного маршруту в графі.

Робота інтерактивної мапи досліджена на сценарії, що охоплює як штатний моніторинг групи, так і тимчасову ізоляцію окремого вузла. Протягом тактів 1-3 усі мобільні апарати мають прямий або транзитний шлях до шлюзу ESP32-GW. Серверна мапа на кожному такті оновлює їхні координати, рівень заряду батареї, активні лінії ретрансляції та наносить поточні сенсорні вимірювання.

На тактах 4-7 вузол ESP32-N4 зміщується у північно-західному напрямку на відстань понад Rmax відносно сусідніх апаратів, унаслідок чого шлях до шлюзу в графі зв’язності переривається. Рухаючись автономно, вузол фіксує підвищені та аномальні умовні показники сенсора (720, 950, 890 та 680 ум. од.). Оскільки сервер не отримує нових пакетів від цього апарата, на інтерактивній мапі (рис. 2, а) маркер вузла 4 відображається сірим кольором із червоним контуром і залишається на останній відомій серверу позиції (з координатами та зарядом станом на 3-й такт), а нові точки вимірювань у північно-західному секторі не відображаються. У локальному буфері вузла за цей час накопичуються 4 записи.

На 8-му такті вузол ESP32-N2 переміщується ближче до північно-західного сектора, утворюючи транзитний маршрут ESP32-GW ↔ ESP32-N2 ↔ ESP32-N4. Після появи маршруту на мапі оновлюються актуальні координати й заряд вузла ESP32-N4, а вздовж пройденої ним ділянки наносяться збережені сенсорні вимірювання за такти 4-7. У спливаючому вікні кожного запису виводиться час фактичного вимірювання, час отримання даних сервером та тривалість затримки передавання в тактах (рис. 2, б).



Рисунок 2. – Відображення стану mesh-мережі
на інтерактивній мапі:
а) активні радіолінки доступних вузлів та фіксація
останньої відомої серверу позиції для недосяжного вузла
ESP32-N4;
б) відновлено маршрут через вузол 2 та нанесено
буферизовані точки вимірювань для вузла 4

Розроблена програмна модель та інтерактивна Web-GIS мапа дозволяють наочно візуалізувати комплексний стан мобільної mesh-мережі у динаміці: просторове розташування та технічний статус апаратів, перебудову маршрутів ретрансляції та розподіл зібраних сенсорних показників. Розмежування фактичного стану вузлів і стану, відомого серверу, а також імітація локальної буферизації та відкладеного пересилання дають змогу зберігати результати вимірювань під час розривів з'єднання й доставляти їх на сервер після відновлення маршруту, за умови збереження даних у буфері та повторної появи доступного шляху до шлюзу.

Подальші дослідження в межах дисертаційної роботи планується спрямувати на уточнення моделі фізичного рівня радіоканалу (з урахуванням RSSI та обмежень пропускної здатності LoRa), розробку методів попередньої обробки та стиснення інформаційно-ємних даних (зокрема з візуальних сенсорів та камер) для передавання через вузькосмугові мережі, а також створення алгоритмів автоматичного перерозподілу завдань між мобільними вузлами у разі затримок.

Список використаних джерел

1. Cotrim J. R., Kleinschmidt J. H. LoRaWAN Mesh Networks: A Review and Classification of Multihop Communication. *Sensors.* 2020. Vol. 20, No. 15. Art. 4273. DOI: 10.3390/s20154273

2. Berto R., Napoletano P., Savi M. A LoRa-Based Mesh Network for Peer-to-Peer Long-Range Communication. *Sensors.* 2021. Vol. 21, No. 13. Art. 4314. DOI: 10.3390/s21134314

3. Solé J. M., Centelles R. P., Freitag F., Meseguer R. Implementation of a LoRa Mesh Library. *IEEE Access.* 2022. Vol. 10. P. 113158–113171. DOI: 10.1109/ACCESS.2022.3217215

4. Koukis G., Safouri G., Tsaoussidis V. All about Delay-Tolerant Networking (DTN) Contributions to Future Internet. *Future Internet.* 2024. Vol. 16, No. 4. Art. 129. DOI: 10.3390/fi16040129

5. Centelles R. P., Meseguer R., Freitag F., Navarro L. A Minimalistic Distance-Vector Routing Protocol for LoRa Mesh Networks. *IEEE Access.* 2024. Vol. 12. P. 128941–128962. DOI: 10.1109/ACCESS.2024.3443605

ПІДСЕКЦІЯ
МОДЕЛІ, МЕТОДИ ТА ЗАСОБИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ

УДК 004.8

Катерина Антіпова
PhD, доцентка (б. в. з.)
кафедри інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ВИЯВЛЕННЯ ЗГЕНЕРОВАНОГО ТЕКСТУ В УМОВАХ ЗМАГАЛЬНИХ АТАК

Зі стрімким розвитком штучного інтелекту великі мовні моделі (large language model, LLM) стають незамінними інструментами як у повсякденному житті, так і в професійній діяльності. Але при цьому викликають занепокоєння щодо неналежного використання згенерованого тексту (AI-generated text, AIGT), зокрема поширення дезінформації, порушення принципів академічної доброчесності та створення шкідливого контенту. Як наслідок, регулювання використання AIGT та ефективне розрізнення його від тексту, написаного людиною (human-written text, HWT), стали критично важливими завданнями. Однак через схожість між AIGT та HWT здатність людини ідентифікувати згенерований текст не перевищує ймовірності випадкового вгадування. Тому існує нагальна потреба в більш досконалих методах виявлення AIGT.

Сучасні детектори AIGT постійно вдосконалюються, але все ще стикаються з численними проблемами. По-перше, ефективність детекторів суттєво знижується через різні модифікації тексту: заміни, перестановки, видалення чи вставки слів, а також зміни на рівні речень, перефразування та змагальні атаки. Такі зміни вводять детектори в оману, оскільки вносять ледь помітні корективи, що значно погіршує точність виявлення. По-друге, робота детекторів часто погіршується при обробці даних, що виходять за межі розподілу навчальної вибірки (out-of-distribution, OOD), наприклад, текстів із предметних сфер, відмінних від тих, що використовувалися для навчання, багатомовних текстів або контенту, створеного іншими LLM. Відмінності в розподілі даних між зразками OOD і навчальним

набором ускладнюють ефективну узагальнювальну здатність моделей. По-третє, детектори зазвичай менш ефективні у виявленні згенерованого контенту у документах, що поєднують текст, написаний людиною, і згенерований текст. Хоча змішаний контент (AI-human text, AHT) є поширеним явищем у повсякденному житті, багато сучасних детекторів розроблено спеціально для ідентифікації виключно AIGT, що суттєво обмежує їхню практичну застосовність. Ці проблеми підкреслюють необхідність створення більш надійних і стійких детекторів AIGT.

Нещодавні дослідження засвідчили, що сучасні детектори AIGT контенту, дуже вразливі до текстових збурень [1]. Навіть незначні зміни, наприклад, заміна слів абревіатурами або просто додавання зайвих пробілів, можуть суттєво вплинути на результати виявлення. Під текстовими збуреннями розуміють різноманітні модифікації початкової форми тексту. Залежно від масштабу змін їх можна поділити на локальні збурення та глобальне перефразування. Локальні збурення зазвичай стосуються невеликих фрагментів тексту й часто залишаються непомітними для людини. Вони переважно полягають у вставці, видаленні, перестановці чи заміні елементів на рівні окремих символів або слів. Хоча такі зміни можуть здаватися людині несуттєвими, вони здатні призвести до помилкової класифікації з боку деяких детекторів. На відміну від локальних збурень, глобальне перефразування може змінювати загальну форму тексту. Перефразування, по суті, передбачає зміну структури семантичних і синтаксичних елементів тексту. У цьому процесі змінюється мовне оформлення, тоді як початковий зміст зберігається. Завдяки здатності генерувати різноманітні варіанти висловлення зі збереженням первинного змісту, перефразування стало ключовою стратегією обходу систем виявлення AIGT. Доведено, що багаторазове перефразування значно знижує ефективність виявлення [2].

Розроблено чимало детекторів для розрізнення суто AIGT і HWT [3, 4], їхня робота ґрунтується на припущенні, що контент походить з одного джерела – або від людини, або від штучного інтелекту. Однак у реальних умовах часто трапляються AHT, наприклад, написані людьми тексти, вдосконалені штучним інтелектом. Дослідження, такі як [5], свідчать, що наявні детектори мають труднощі з ідентифікацією AHT, особливо коли йдеться про незначні модифікації чи стилістичні зміни. Це суттєво обмежує можливості практичного застосування детекторів AIGT. Такі гібридні стилі тексту розмивають межу між характеристиками AIGT та HWT.

Окрім розробки ефективних методів підвищення стійкості детекторів AIGT, не менш важливим є й оцінювання цієї стійкості. Ефективні методи оцінювання допомагають виявити слабкі місця детектора, що дозволяє цілеспрямовано вдосконалювати модель. Ключ до перевірки стійкості детекторів AIGT до текстових збурень полягає у створенні модифікованих зразків. Такі зразки можна генерувати шляхом внесення випадкових змін у дані. Для цього якнайкраще підходять методи аугментації текстових даних. Ці методи зазвичай розширюють навчальні набори даних, покращуючи таким чином ефективність моделі, шляхом заміни слів синонімами, а також випадкових вставок, перестановок і видалень слів. У цьому контексті такі методи можна адаптувати для генерування змінених зразків з метою оцінювання стійкості детектора. Однак, оскільки ці методи за своєю суттю є стохастичними, вони можуть вносити певні зміщення в результати оцінювання.

Методи генерування змагальних збурень (adversarial perturbations) відіграють вирішальну роль в оцінюванні стійкості моделей. Використовуючи вразливості детектора, ці методи дозволяють генерувати змагальні зразки (adversarial samples) у високоефективний і цілеспрямований спосіб. С. Рен та ін. [6] запропонували алгоритм заміни слів синонімами, що ґрунтується на семантиці слів і ймовірності класифікації. Їхній метод дозволяє мінімізувати точність класифікації за умови дуже низької частоти заміни слів. З. Чжао та ін. [7] генерували змагальні збурення на рівні речень за допомогою генеративно-змагальних мереж, створюючи природні, зручні для читання зразки, здатні вводити класифікатори в оману.

Генерування перефразувань може використовуватися для оцінювання стійкості систем виявлення AIGT, до атак шляхом перефразування. Цей процес передбачає переписування заданого тексту в іншій формі зі збереженням його початкового змісту. Для створення таких зразків доступно чимало спеціалізованих інструментів перефразування [8, 9]. Зазвичай вони базуються на попередньо навчених моделях типу sequence-to-sequence. Крім того, перефразування можна здійснювати за допомогою промптів у поєднанні з LLM. Генерування текстів у різних стилях дозволяє перевірити стійкість детектора до перефразованого тексту.

Методи атаки на детектори AIGT можна поділити на два класи: такі, що базуються на запитах, і такі, що не потребують запитів. Методи першого типу надсилають запити до цільової моделі для отримання вихідних даних і коригування стратегії атаки. До них належать:

- PWWS [6]. Метод атаки типу «чорна скринька», спрямований на моделі класифікації тексту. Він обчислює та зважує значущість кожного слова в реченні й замінює слова з високою значущістю синонімами з бази WordNet. Це дозволяє створювати змагальні зразки, які вводять модель в оману, зберігаючи при цьому початковий зміст.
- TextFooler [10]. Метод атаки типу «чорна скринька», призначений для завдань класифікації тексту та логічного висновку на основі природної мови. Він оцінює слова за ступенем їхнього впливу на прогноз моделі та замінює їх семантично близькими й граматично правильними синонімами. У результаті створюються зразки для атаки, що зберігають зміст і структуру речення.
- BERTAttack [11]. Метод атаки типу «чорна скринька», що генерує зразки для атаки за допомогою попередньо навченої моделі BERT. Він замінює певні слова в цільовому реченні словами-кандидатами, які мають високу ймовірність появи згідно з моделлю BERT. Це призводить до створення зразків, які вводять модель в оману, зберігаючи семантичну та граматичну схожість із початковим реченням.
- HQA [12]. Метод атаки типу «чорна скринька» для завдань класифікації тексту. Він ініціалізує зразки для атаки шляхом виявлення збурень, що мінімально впливають на семантичну узгодженість, а потім ітеративно замінює слова, використовуючи набори синонімів-кандидатів. Це дозволяє генерувати високоякісні тексти для атаки зі значно меншою кількістю запитів порівняно зі звичайними методами пошуку.
- ABP [13]. Метод атаки типу «чорна скринька», що використовує попередні знання для ефективного керування заміною слів. Він застосовує метод бустингу для оцінки важливості слів і пропонує дві стратегії, ефективні з погляду кількості запитів: ABPfree (атака без запитів) та ABPguide (атака з керованим пошуком). Ці методи значно скорочують кількість запитів, забезпечуючи при цьому високий рівень успішності атаки.
- FastTextDodger [14]. Метод атаки типу «чорна скринька», розроблений для створення високоякісних текстів для атаки з мінімальною кількістю запитів. Він визначає відповідні заміни слів, використовуючи семантичні представлення з моделі вбудовувань FastText. Це дозволяє отримувати зразки для атаки, які є граматично правильними та семантично узгодженими з початковими реченнями.

Методи «нульових запитів» (zero-query) здійснюють атаки без отримання будь-якої інформації від цільової моделі. До таких методів належать:

- WordNet [15]. Метод атаки типу «чорна скринька», що ґрунтується на лексичній заміні. Він замінює вибрані слова синонімами з лексичної бази даних WordNet, не потребуючи доступу до цільового детектора чи його градієнтів. Варіанти заміни обмежуються вимогами щодо сумісності частин мови та семантичної подібності, щоб зберегти початковий зміст і граматичну правильність тексту.
- T-PGD [16]. Метод атаки типу «чорна скринька», що поширює методи атак на основі оптимізації (запозичені з комп'ютерного зору) на сферу обробки природної мови. Він вносить збурення на рівень вбудовувань, посилює їх шляхом прямого поширення сигналу, а згодом використовує масковану мовну модель для декодування змагальних прикладів, отриманих без запитів до цільової системи.
- HMGC [15]. Фреймворк типу «чорна скринька» для атак із нульовою кількістю запитів. Він апроксимує детектор за допомогою сурогатної моделі, оцінює слова на основі градієнтної чутливості та перплексії, а також замінює найбільш значущі слова за допомогою маскованої мовної моделі на базі кодера. Обмеження забезпечують плавність і семантичну узгодженість, тоді як динамічне змагальне навчання вдосконалює стратегію атаки.
- Перефразування. Метод атаки з нульовою кількістю запитів, що генерує змагальні приклади шляхом переписування тексту на рівні речень і абзаців зі збереженням початкового змісту. Він використовує модель перефразування DIPPER для зміни порядку, об'єднання та розділення речень, створюючи лексично й синтаксично різноманітні варіанти тексту, які зберігають семантичну цілісність.
- Перефразування промптів. Метод атаки з нульовою кількістю запитів, що передбачає переписування промптів, які використовуються для генерації тексту штучним інтелектом, замість модифікації самого згенерованого тексту. Для реконструкції вхідних промптів використовується модель перефразування Pegasus. Згенеровані результати зберігають початковий зміст, демонструючи при цьому більшу стилістичну варіативність.
- Зворотний переклад. Атака на рівні абзацу, що полягає в перекладі тексту іншою мовою з подальшим перекладом назад на мову оригіналу. Цей метод справляє сильний ефект нормалізації, часто порівнянний із глибоким перефразуванням.

Щодо різних методів зловмисних атак, спостерігається чіткий компроміс між ефективністю атаки та якістю тексту. HQA досягає найвищого успіху завдяки агресивній модифікації вхідних даних, що призводить до вищого рівня збурень і зростання перплексії. Натомість

BERTAttack забезпечує найбільш оптимальний баланс: високий показник успішності атаки досягається при зміні лише 10% токенів і збереженні найвищого рівня семантичної схожості. Це свідчить про те, що заміна слів на основі контекстуальних мовних моделей є особливо ефективною для створення непомітних змагальних прикладів. Методи ABP і FastTextDodger також демонструють високу ефективність, хоча й потребують більше модифікацій порівняно з BERTAttack. Серед методів, що не потребують запитів до моделі, T-PGD є найпотужнішою атакою за показником успішності атаки, однак вона суттєво погіршує плавність тексту та його семантичну схожість із оригіналом. Простіші підходи, що базуються на трансформаціях, як-от заміна слів синонімами з WordNet чи зворотний переклад, добре зберігають якість тексту, але не забезпечують стабільного обходу детектора.

Список використаних джерел

1. Liu X., Li Y., & Li K. (2025). Enhancing the robustness of AI-generated text detectors: A survey. *Mathematics,* 13(13), Article 2145. DOI: https://doi.org/10.3390/math13132145
2. Sadasivan V. S., Kumar A., Balasubramanian S., Wang W., & Feizi S. (2023). Can AI-generated text be reliably detected? arXiv. DOI: https://doi.org/10.48550/arXiv.2303.11156
3. Su J., Zhuo T. Y., Wang D., & Nakov P. (2023). DetectLLM: Leveraging log rank information for zero-shot detection of machine-generated text. *Findings of the Association for Computational Linguistics:* EMNLP 2023, 12395–12412. DOI: https://doi.org/10.18653/v1/2023.findings-emnlp.827
4. Bao G., Zhao Y., Teng Z., Yang L., & Zhang Y. (2024). Fast-DetectGPT: Efficient zero-shot detection of machine-generated text via conditional probability curvature. *International Conference on Learning Representations (ICLR).* DOI: https://doi.org/10.48550/arXiv.2310.05130
5. Zhang Q., Gao C., Chen D., Huang Y., Huang Y., Sun Z., Zhang S., Li W., Fu Z., Wan Y., & Sun L. (2024). LLM-as-a-coauthor: Can mixed human-written and machine-generated text be detected? *Findings of the Association for Computational Linguistics: NAACL 2024*, 409–436. DOI: https://doi.org/10.18653/v1/2024.findings-naacl.29
6. Ren S., Deng Y., He K., & Che W. (2019). Generating natural language adversarial examples through probability weighted word saliency. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1085–1097. DOI: https://doi.org/10.18653/v1/P19-1103

7. Zhao Z., Dua D., & Singh S. (2018). Generating natural adversarial examples. *International Conference on Learning Representations (ICLR).* DOI: https://doi.org/10.48550/arXiv.1710.11342

8. Razaq A., Shah B., Khan G., Alfandi O., Ullah A., Halim Z., & Rahman A. U. (2025). Improving paraphrase generation using supervised neural-based statistical machine translation framework. *Neural Computing and Applications,* 37, 7705–7719. DOI: https://doi.org/10.1007/s00521-023-08830-4

9. Ouahrani L., & Bennouar D. (2024). Paraphrase generation and supervised learning for improved automatic short answer grading. *International Journal of Artificial Intelligence in Education,* 34, 1627–1670. https://doi.org/10.1007/s40593-023-00391-w

10. Jin D., Jin Z., Zhou J. T., & Szolovits P. (2020). Is BERT really robust? A strong baseline for natural language attack on text classification and entailment. *Proceedings of the AAAI Conference on Artificial Intelligence,* 34, 8018–8025. DOI: https://doi.org/10.48550/arXiv.1907.11932

11. Li L., Ma R., Guo X., Wang Q., Qiu X., & Tang X. (2020). BERT-Attack: Adversarial attack against BERT using BERT. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 6193–6202. DOI: https://doi.org/10.48550/arXiv.2004.09984

12. Liu H., Xu Z., Zhang X., Zhang F., Ma F., Chen H., Yu H., & Zhang X. (2024). HQA-Attack: Toward high quality black-box hard-label adversarial attack on text. *Advances in Neural Information Processing Systems,* 36. DOI: https://doi.org/10.48550/arXiv.2402.01806

13. Yu, Z., Chen, Z., & He, K. (2024). Query-efficient textual adversarial example generation for black-box attacks. *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 556–569. DOI: https://doi.org/10.18653/v1/2024.naacl-long.31

14. Hu X., Liu G., Zheng B., Zhao L., Wang Q., Zhang Y., & Du M. (2024). FastTextDodger: Decision-based adversarial attack against black-box NLP models with extremely high efficiency. *IEEE Transactions on Information Forensics and Security*. DOI: https://doi.org/10.1109/TIFS.2024.3350376

15. Zhou Y., He B., & Sun L. (2024). Humanizing machine-generated content: Evading AI-text detection through adversarial attack. *Proceedings of the Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 8427–8437. DOI: https://doi.org/10.48550/arXiv.2404.01907

16. Yuan L., Zhang Y., Chen Y., & Wei W. (2023). Bridge the gap between CV and NLP! A gradient-based textual adversarial attack framework. *Findings of the Association for Computational Linguistics:* ACL 2023, 7132–7146. DOI: https://doi.org/10.18653/v1/2023.findings-acl.446

УДК 004.89:004.77

Єгор Білий
здобувач першого (бакалаврського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Світлана Боровльова
старша викладачка кафедри
інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

РОЗРОБКА НАВЧАЛЬНОЇ ІТ-ПЛАТФОРМИ З ВИКОРИСТАННЯМ LLM

Сучасна система вищої освіти стикається з необхідністю постійної адаптації до швидкого розвитку інформаційних технологій. Традиційні системи управління навчанням забезпечують базовий функціонал для розміщення матеріалів та тестування, проте вони не здатні надавати персоналізовану допомогу студентам у режимі реального часу.

Метою даної роботи є розробка мікросервісної ІТ-платформи з інтеграцією великих мовних моделей для автоматизації освітніх процесів та забезпечення адаптивного навчання. Основна ідея проєкту полягає у використанні штучного інтелекту в ролі віртуального асистента та ментора. Така система дозволяє не лише надавати навчальний контент, але й автоматично генерувати тестові завдання, формувати розгорнуті пояснення помилок та підлаштовувати складність матеріалу під індивідуальні потреби студента.

Однією з головних проблем використання базових моделей LLM є явище «галюцинацій» та відсутність доступу до специфічної локальної інформації (наприклад, документацій та книжок). Для усунення цього недоліку в платформі застосовується технологія RAG – генерація, доповнена пошуком. Процес полягає у тому, що перед формуванням відповіді система спочатку здійснює семантичний пошук релевантних фрагментів тексту у власній базі знань. Знайдений контекст додається до запиту, що змушує мовну модель генерувати відповідь виключно на основі перевірених матеріалів.

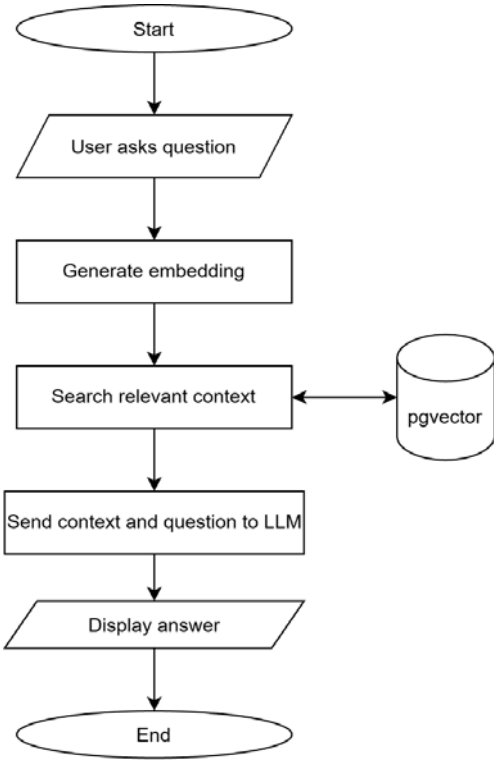


Рисунок 1. – Схема алгоритму функціонування RAG-модуля

Архітектура проєкту базується на сучасному мікросервісному підході, що забезпечує високу масштабованість, надійність та зручність у подальшій підтримці. Клієнтська частина реалізується за допомогою фреймворку Next.js. Вибір цієї технології зумовлений підтримкою серверного рендерингу, що забезпечує високу швидкість завантаження сторінок та плавну взаємодію з користувачем завдяки екосистемі React.

Серверна частина побудована на базі FastAPI – сучасного вебфреймворку для створення API на Python. Python є стандартом у сфері штучного інтелекту, що дозволяє безшовно інтегрувати такі фреймворки, як LangChain, для гнучкого управління робочими процесами LLM. FastAPI забезпечує високу продуктивність завдяки асинхронній обробці запитів, що є критично важливим при очікуванні

відповідей від зовнішніх API мовних моделей (наприклад, OpenAI або Anthropic).

Для зберігання даних використовується реляційна система управління базами даних PostgreSQL. Окрім традиційного збереження інформації про користувачів, курси та результати тестувань, PostgreSQL використовується як векторне сховище завдяки розширенню pgvector. Це розширення дозволяє зберігати векторні представлення текстів та виконувати операції семантичного пошуку найближчих сусідів (наприклад, cosine similarity). Використання єдиної бази даних для реляційних та векторних даних значно спрощує інфраструктуру проєкту та знижує витрати на обслуговування у порівнянні з використанням окремих спеціалізованих векторних баз даних.

Для забезпечення стабільності та зручності розгортання усі компоненти системи контейнеризуються за допомогою Docker та налаштовуються через Docker Compose. Фізичне розгортання та хостинг платформи здійснюватиметься у хмарному середовищі AWS.

Використання інфраструктури AWS дозволить забезпечити високу доступність системи, гнучке масштабування обчислювальних ресурсів під час пікових навантажень (наприклад, під час великої кількості одночасного проходження тестів) та надійний захист академічних даних.

Отже, розробка навчальної IT-платформи з використанням LLM та RAG на базі стеку Next.js, FastAPI, PostgreSQL та розгортанням на AWS є актуальним та перспективним інженерним рішенням. Впровадження такої системи підвищить ефективність засвоєння матеріалу студентами завдяки персоналізованому підходу та значно знизить навантаження на викладачів, автоматизувавши процеси модерації та генерації навчального контенту.

Список використаних джерел

1. Python Documentation. URL: https://docs.python.org/3/ (дата звернення: 14.10.2026).
2. FastAPI Documentation. URL: https://fastapi.tiangolo.com/ (дата звернення: 14.10.2026).
3. pgvector: Open-source vector similarity search for Postgres. URL: https://github.com/pgvector/pgvector (дата звернення: 14.10.2026).
4. OpenAI API Platform Documentation. URL: https://developers.openai.com/api/docs (дата звернення: 14.10.2026).
5. Гриценко Ю. Що таке RAG і як його можна використовувати на практиці? URL: https://blog.ithillel.ua/articles/what-is-rag (дата звернення: 14.10.2026).

УДК 004.78:004.41

Владислав Блідар
здобувач першого (бакалаврського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євгеній Стоєв
старший викладач кафедри
інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

АРХІТЕКТУРА КІМНАТНОГО (ROOM-BASED) МАРШРУТИЗАТОРА ПОВІДОМЛЕНЬ ДЛЯ ВЕБСОКЕТ-СЕРВЕРІВ: ПРОБЛЕМИ МАСШТАБУВАННЯ ТА ШЛЯХИ ЇХ ВИРІШЕННЯ

У сучасних вебзастосунках, зокрема на платформах для спільного прослуховування медіаконтенту та колаборації, виникає гостра потреба у синхронізації стану між клієнтами в реальному часі. Використання класичного протоколу HTTP для таких завдань є неефективним через високі накладні витрати на встановлення з'єднань і неможливість ініціювати передавання даних з боку сервера, тому стандартом де-факто став протокол WebSocket. Проте базова реалізація широкомовної розсилки повідомлень стикається з серйозними обмеженнями при масштабуванні, коли користувачів необхідно ізолювати у віртуальні кімнати або хаби. Наївний підхід, що передбачає зберігання масивів підключень для кожної кімнати, призводить до деградації продуктивності, витоків пам'яті через неактивні з'єднання та складнощів при горизонтальному масштабуванні бекенду. З одного боку, розробники потребують простого API для роботи з кімнатами, з іншого – система має залишатися стабільною за умов тисяч одночасних підключень і нестабільних мереж користувачів.

Для вирішення цих проблем застосовується оптимізована архітектура кімнатного маршрутизатора повідомлень, побудована на базі високопродуктивних фреймворків, таких як Fastify. Можна виокремити кілька ключових складників цієї архітектури.

Двостороннє індексування підключень: основною ідеєю є відмова від лінійних структур даних на користь хеш-таблиць. Сервер

підтримує дві пов’язані структури: відображення ідентифікатора кімнати на множину активних сокетів та відображення ідентифікатора сокета на множину кімнат, у яких він перебуває. Такий підхід забезпечує константний (у середньому) час виконання операцій пошуку, додавання та видалення клієнтів, а зворотний індекс дозволяє за одну операцію очистити всі підписки сокета при його відключенні. Вартість розсилки при цьому залежить лише від кількості учасників конкретної кімнати, а не від загальної кількості підключень на сервері.

Керування життєвим циклом кімнат: кімната існує лише доти, доки в ній є хоча б один учасник. Після відключення останнього клієнта відповідні записи явно видаляються з індексів, а звільнення пам'яті виконує збирач сміття. Це запобігає накопиченню «порожніх» кімнат і вичерпанню оперативної пам'яті сервера під час тривалої роботи.

Горизонтальне масштабування через брокер повідомлень: оскільки високонавантажені додатки вимагають розгортання кількох екземплярів сервера у контейнеризованих середовищах, локальна маршрутизація доповнюється розподіленим брокером. Використання Redis Pub/Sub дозволяє зв’язати ізольовані вузли WebSocket-сервера в єдину мережу. Коли подія, наприклад зміна треку чи оновлення стану кімнати, генерується на одному сервері, повідомлення публікується у відповідний канал Redis. Кожен екземпляр підписується лише на ті канали, клієнти яких фізично підключені до нього, і ретранслює отримані дані виключно цільовим користувачам. Це мінімізує внутрішній мережевий трафік і знижує навантаження на центральний процесор. Водночас слід враховувати те, що Redis Pub/Sub гарантує доставку за принципом «щонайбільше один раз», тому критичний стан кімнати доцільно зберігати в самому Redis і передавати клієнтам повторно під час перепідключення.

Стійкість до розривів мережі: для виявлення розірваних TCP-з’єднань запроваджується механізм контролю на базі обміну пакетами Ping/Pong (heartbeat). У разі втрати зв’язку сокет не видаляється з кімнати миттєво; маршрутизатор ініціює тимчасову затримку очікування (grace period). Якщо клієнт успішно перепідключається протягом цього вікна, його стан відновлюється без генерації надлишкових подій входу та виходу для інших учасників кімнати. Такий підхід особливо важливий для мобільних клієнтів, які часто короткочасно втрачають з’єднання під час перемикання між мережами.

Авторизація та ізоляція кімнат: перевірка прав доступу виконується на етапі встановлення з’єднання (валідація токена) та

щоразу, коли клієнт намагається приєднатися до кімнати. Маршрутизатор гарантує, що повідомлення потрапляють лише до учасників відповідної кімнати, а клієнт не може надсилати події в кімнати, до яких не приєднаний. Це усуває клас вразливостей, пов'язаних з несанкціонованим доступом до чужого стану (IDOR).

Синхронізація стану відтворення: у сценарії спільного прослуховування важливо, щоб учасники чули один і той самий фрагмент треку одночасно. Замість постійної передачі позиції відтворення кожному клієнту сервер зберігає опорний стан кімнати (поточний трек, позицію та мітку часу її фіксації), а клієнти самостійно обчислюють актуальну позицію. Це зменшує кількість службових повідомлень і зберігає узгодженість станів навіть за наявності мережевих затримок.

Захист від перевантаження: до маршрутизатора додаються обмеження частоти повідомлень від одного клієнта (rate limiting), максимального розміру повідомлення та контроль черги вихідних даних (backpressure). Повільні одержувачі не блокують розсилку іншим учасникам кімнати, а за критичного відставання їхнє з'єднання коректно закривається.

Спостережуваність і тестування: для експлуатації системи в реальних умовах важливо збирати метрики кількості з'єднань, кімнат і затримки доставки повідомлень, а також перевіряти поведінку маршрутизатора навантажувальними тестами з імітацією розривів мережі.

Впровадження описаної архітектури кімнатних хабів дозволяє досягти стабільної роботи додатку при тисячах одночасних з’єднань. Заміна лінійного пошуку на хеш-таблиці усуває просадки продуктивності при масових розсилках, а делегування міжсерверної комунікації на Redis забезпечує горизонтальне масштабування без зміни клієнтського коду. Комплексний підхід до управління життєвим циклом з’єднань формує надійний фундамент для побудови інтерактивних мультимедійних платформ, гарантуючи мінімальну затримку доставки повідомлень та раціональне використання апаратних ресурсів.

Список використаних джерел

1. Fette I., Melnikov A. The WebSocket Protocol : RFC 6455. Internet Engineering Task Force (IETF). 2011. URL: https://datatracker.ietf.org/doc/html/rfc6455 (дата звернення: 19.10.2026).
2. Fastify Reference Documentation & Architecture / Fastify Team. URL: https://fastify.dev/docs/latest/ (дата звернення: 19.10.2026).

3. Redis Pub/Sub and Distributed Messaging / Redis Documentation. URL: https://redis.io/docs/interact/pubsub/ (дата звернення: 19.10.2026).
4. Kleppmann M. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. Sebastopol : O'Reilly Media, 2017. 616 p.
5. Глибовець М. М., Олецький О. В. Розподілені системи та паралельні обчислення : навч. посіб. Київ : НаУКМА, 2018. 210 с.

УДК 004.8:519.2

Олександр Горелов
здобувач другого (магістерського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Марина Фаленкова
старша викладачка кафедри
інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

СИСТЕМА ПРОГНОЗУВАННЯ РИЗИКУ НА ОСНОВІ АНСАМБЛЕВИХ МОДЕЛЕЙ

В умовах цифровізації та зростання обсягів інформації актуальним завданням є розроблення програмних засобів автоматизованого аналізу даних і прогнозування ризиків. Такі системи допомагають виявляти закономірності, оцінювати ймовірність небажаних подій і підтримувати прийняття обґрунтованих рішень. Прогнозування ризиків застосовується у фінансовій сфері, кредитуванні, інформаційній безпеці та управлінні виробничими процесами, де помилки прогнозу можуть призводити до суттєвих втрат [1, 2]. Якість результатів залежить від характеристик даних, вибраного алгоритму та методики оцінювання.

Перспективним напрямом розв'язання означеного завдання є ансамблеве машинне навчання, яке передбачає об'єднання прогнозів декількох базових моделей. Такий підхід дає змогу використовувати переваги різних алгоритмів, зменшувати вплив окремих помилок і покращувати узагальнювальну здатність системи [3, 4].

Метою даної роботи є підвищення ефективності прогнозування ризику шляхом побудови та порівняння ансамблевих моделей Random Forest, AdaBoost і XGBoost.

Для досягнення поставленої мети передбачено аналіз сучасних підходів до прогнозування ризиків, формалізацію задачі класифікації, визначення структури програмної системи, вибір методів машинного навчання та показників оцінювання. Система має забезпечувати підготовку вхідних даних, навчання моделей, формування прогнозів і порівняння результатів за однакових експериментальних умов.

Задачу прогнозування ризику можна представити у вигляді навчального набору даних:

$$D = (x_i, y_i)_{i=1}^{N}, \quad (1)$$

де: N – кількість спостережень;
x_i – вектор ознак об'єкта;
y_i – цільове значення.

Для бінарної класифікації $y_i \in \{0, 1\}$, де клас 1 відповідає наявності ризикової події, а клас 0 – її відсутності. Навчена модель оцінює ймовірність належності об'єкта до ризикового класу, після чого за встановленим порогом формується кінцевий прогноз. Вибір порогу залежить від особливостей предметної області та співвідношення наслідків хибно-позитивних і хибно-негативних рішень.

Одним із досліджуваних алгоритмів є Random Forest, який формує ансамбль дерев рішень, побудованих на випадкових вибірках спостережень і підмножинах ознак. Підсумковий результат визначається голосуванням дерев або усередненням їхніх оцінок імовірності. Перевагами методу є здатність моделювати нелінійні залежності, стійкість до варіацій даних і можливість оцінювання важливості ознак [3]. Водночас збільшення кількості дерев може підвищувати обчислювальні витрати, а складність інтерпретації сукупного прогнозу ускладнює пояснення окремих рішень.

AdaBoost належить до послідовних методів ансамблювання. Під час навчання кожен наступний базовий класифікатор приділяє більше уваги спостереженням, які попередні моделі класифікували неправильно. Внесок окремих класифікаторів у загальний прогноз залежить від їхньої якості. Завдяки цьому алгоритм може покращувати результати відносно простих базових моделей. Проте його особливістю є чутливість до шуму та помилково розмічених даних,

оскільки складні для класифікації спостереження можуть отримувати підвищену вагу [5].

XGBoost реалізує градієнтний бустинг дерев рішень, за якого нові дерева послідовно коригують помилки вже побудованого ансамблю. Важливою особливістю алгоритму є використання регуляризації, що обмежує складність моделі та допомагає зменшити ризик перенавчання. Загальний принцип оптимізації можна представити функцією:

$$\mathcal{L} = \sum_{i=1}^{N} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k), \quad (2)$$

де: $l(y_i, \hat{y}_i)$ – функція втрат, що характеризує помилку прогнозування;

f_k – окреме дерево ансамблю;

K – кількість дерев;

$\Omega(f_k)$ – регуляризаційний доданок, який враховує складність дерева.

Ефективність XGBoost залежить від глибини дерев, швидкості навчання, кількості ітерацій та параметрів регуляризації [6]. Ці характеристики дають змогу гнучко налаштовувати модель відповідно до властивостей набору даних.

Коректне порівняння алгоритмів потребує належної підготовки даних. На цьому етапі перевіряються пропущені значення, дублікати, некоректні записи, типи ознак і співвідношення класів. Після очищення дані поділяються на навчальну й тестову вибірки. Навчальна вибірка використовується для побудови моделей і підбору гіперпараметрів, тоді як тестова – для незалежного оцінювання кінцевих результатів. Операції попередньої обробки, відбору ознак і балансування класів мають налаштовуватися лише на навчальних даних, щоб уникнути витоку інформації. Для підвищення надійності вибору параметрів доцільно застосовувати стратифіковану перехресну перевірку, яка зберігає приблизне співвідношення класів у навчальних блоках.

Якість прогнозування пропонується оцінювати за метриками Accuracy, Precision, Recall, F1-score і ROC-AUC. Accuracy характеризує частку правильних прогнозів, Precision – точність виявлення позитивного класу, а Recall – повноту його виявлення. Для узгодженого оцінювання Precision і Recall використовується F1-score, який визначається як їхнє гармонійне середнє:

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (3)$$

Цей показник є особливо корисним за нерівномірного розподілу класів, коли важливо враховувати як пропущені ризикові події, так і хибні спрацювання. Додатково використовується ROC-AUC – площа під ROC-кривою, яка відображає здатність моделі розрізняти класи за різних порогів класифікації. Її можна представити як:

$$ROC\text{-}AUC = \int_0^1 TPR(FPR)\,, dFPR, \quad (4)$$

де: *TPR* – частка правильно виявлених позитивних випадків;

FPR – частка негативних випадків, помилково віднесених до позитивного класу [7].

Комплексне використання зазначених метрик дає змогу оцінити різні аспекти якості прогнозування та уникнути вибору моделі лише за одним показником.

Практичне значення роботи полягає у формуванні основи програмної системи, яка автоматизує підготовку даних, навчання трьох ансамблевих моделей, формування прогнозів і порівняння їхніх характеристик. Використання Python, бібліотеки Scikit-learn і засобів XGBoost забезпечує реалізацію алгоритмів та обчислення показників якості. Для об'єктивності експерименту необхідно забезпечити однакові умови розбиття даних і узгоджену процедуру оцінювання. Вибір найефективнішого алгоритму має ґрунтуватися на отриманих результатах, а також враховувати обчислювальні витрати й особливості предметної області. Наприклад, у ситуаціях, коли пропуск ризикової події має значні наслідки, пріоритетним може бути Recall, тоді як за необхідності збалансувати повноту й точність виявлення доцільно враховувати F1-score.

Отже, ансамблевий підхід є перспективною основою для розроблення системи прогнозування ризику. Порівняння Random Forest, AdaBoost і XGBoost за узгодженою методикою дасть змогу визначити переваги й обмеження кожного алгоритму та обґрунтувати вибір моделі для подальшого використання. Результати дослідження можуть бути використані для створення програмних засобів підтримки прийняття рішень, спрямованих на своєчасне виявлення ризикових ситуацій і підвищення обґрунтованості прогнозів.

Список використаних джерел

1. Machine learning-driven credit risk: a systemic review. *Neural Computing and Applications*. 2023.

2. Big data in financial risk management: evidence, advances, and open questions: a systematic review. *Frontiers in Artificial Intelligence*. 2025.

3. Breiman L. Random Forests. *Machine Learning*. 2001. Vol. 45. P. 5–32.

4. Fan Z., Yu Z., Yang K. et al. Diverse Models, United Goal: A Comprehensive Survey of Ensemble Learning. *CAAI Transactions on Intelligence Technology*. 2025. Vol. 10, No. 4. P. 959–982.

5. Freund Y., Schapire R. E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*. 1997. Vol. 55, No. 1. P. 119–139.

6. Chen T., Guestrin C. XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016. P. 785–794.

7. Fawcett T. An introduction to ROC analysis. *Pattern Recognition Letters*. 2006. Vol. 27, No. 8. P. 861–874.

УДК 004.8:004.41

Андрій Євсєєв
здобувач другого (магістерського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євген Давиденко
кандидат технічних наук, доцент,
завідувач кафедри інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ВИКОРИСТАННЯ МАШИННОГО НАВЧАННЯ ТА МНЕМОНІЧНОГО МЕТОДУ ХЕЙЗІГА ДЛЯ РОЗРОБКИ МОБІЛЬНОГО ЗАСТОСУНКУ ВИВЧЕННЯ КАНДЗІ

Японська мова належить до найскладніших для опанування мов світу, що зумовлено насамперед її системою письма. Ієрогліфи (кандзі) є логографічними символами, кожен з яких має власне значення, одне або кілька онних (китайських) та кунних (японських) читань, а також суворо визначений порядок написання рисок. Традиційні методи механічного заучування є малоефективними для довготривалої пам'яті, оскільки не створюють стійких асоціативних зв'язків [2, 5, 10]. Сучасні мобільні застосунки частково вирішують цю проблему, проте більшість із них не поєднують мнемонічні техніки, науково обґрунтоване інтервальне повторення та автоматичну перевірку написання в єдиному середовищі.

Метою даної роботи є підвищення ефективності вивчення кандзі шляхом розробки мобільної інтелектуальної системи, що інтегрує мнемонічний метод Джеймса Хейзіга, систему інтервального повторення SM-2 та згорткову нейронну мережу (CNN) для динамічного розпізнавання рукописного введення.

Метод, запропонований Джеймсом Хейзігом у книзі "Remembering the Kanji" [1], кардинально відрізняється від традиційного підходу, який вимагає одночасного заучування форми, значення та читань. Дж. Хейзіг пропонує розділити ці процеси. На першому етапі вивчається лише значення та написання ієрогліфа, а читання ігноруються. Це дозволяє знизити когнітивне навантаження та уникнути плутанини.

Основою методу є принцип декомпозиції. Складні ієрогліфи розкладаються на простіші, неподільні елементи – примітиви (або компоненти). Наприклад, кандзі 時 (час) складається з 日 (сонце/день), 土 (земля) та 寸 (міра довжини). Застосунок зберігає ці історії, а також порядок компонентів та їхню роль у мнемоніці. Це дозволяє користувачу не просто заучувати форму, а розуміти її логіку.

Другим ключовим елементом є створення асоціативних історій. Кожному примітиву присвоюється певний образ або значення. Коли ці образи поєднуються в одному ієрогліфі, користувач створює яскраву, часто абсурдну або емоційно забарвлену історію. Наприклад, кандзі 休 (відпочинок) складається з 人 (людина) та 木 (дерево). Мнемонічна історія: «Людина стоїть біля дерева, щоб відпочити». Дослідження в галузі когнітивної психології підтверджують, що такий підхід активує образне мислення (dual coding theory) [5] та створює значно міцніші нейронні зв'язки, ніж ізольовані факти.

Додатковим інноваційним рішенням у застосунку є впровадження нескінченного просторового полотна для малювання. Замість жорсткої сітки, яка створює психологічний тиск та страх помилки, користувач може розміщувати ієрогліфи у будь-якій частині екрана, під різними кутами або навіть догори дриґом. Це рішення спирається на метод локусів (Memory Palace), який широко використовується в мнемотехніці [5, 10]. Створюючи власні просторові кластери та розташовуючи символи у незвичних орієнтаціях, користувач формує додаткові просторові «якорі» для довготривалої пам’яті. Крім того, вільне малювання без обмежень розвиває моторну гнучкість та долає ефект завченої безпорадності перед ідеальним зразком, роблячи процес навчання більш творчим.

Для запобігання кривій забування Германа Еббінгауза [2] використовується алгоритм SM-2 [3]. Кожна картка має стан, що включає кількість успішних повторень, фактор легкості та поточний інтервал у днях. Користувач оцінює складність пригадування за шкалою від 0 до 3, після чого система автоматично розраховує оптимальний інтервал до наступного показу [3]. Якщо користувач припускається помилки, прогрес скидається, і картка повертається в активну ротацію. Це забезпечує адаптивність навчального процесу під індивідуальні особливості пам’яті користувача.

Особливістю застосунку є модуль автоматичної перевірки написання. Користувач малює ієрогліф на сенсорному екрані, а система аналізує зображення за допомогою згорткової нейронної мережі [4]. На відміну від класичних алгоритмів порівняння траєкторій (наприклад, Dynamic Time Warping), CNN є стійкою до варіативності

почерку, нахилу та товщини ліній. Мережа складається з шарів згортки та підвибірки, які виокремлюють ключові ознаки символу. Для забезпечення навчального процесу база кандзі та векторні зображення для анімації штрихів завантажуються з відкритих джерел [6, 7]. Для забезпечення роботи в офлайн-режимі та низької затримки модель тренується на PyTorch [11], після чого конвертується у формат TFLite і виконується безпосередньо на пристрої користувача без звернення до сервера.

Система побудована за клієнт-серверною архітектурою, що забезпечує роботу в офлайн-режимі та синхронізацію даних між пристроями. Клієнтська частина відповідає за інтерфейс користувача та локальне збереження даних, серверна – за аутентифікацію, збереження прогресу та обмін даними [8, 9]. Синхронізація здійснюється через програмний інтерфейс із використанням стратегії розв’язання конфліктів на основі часу останньої зміни. Крім базового навчання, система надає можливість створення персоналізованих навчальних наборів, перегляду детальної статистики прогресу та підтримує різні рівні складності.

Отже, результатом роботи є мобільна система, що поєднує педагогічні методики та сучасні технології штучного інтелекту. За результатами розгляду можна зробити такі висновки:

– поєднання мнемонічного методу Хейзіга та алгоритму інтервального повторення SM-2 забезпечує міцне засвоєння матеріалу та знижує когнітивне навантаження;

– використання згорткових нейронних мереж дозволяє автоматизувати перевірку правильності написання кандзі з урахуванням індивідуальних особливостей почерку;

– інтеграція нескінченного просторового полотна на основі методу локусів дозволяє створювати персоналізовані просторові асоціації, знижує стрес від перфекціонізму та розвиває моторну гнучкість;

– інтеграція моделі у форматі TFLite забезпечує миттєвий зворотний зв'язок без звернення до сервера, що критично важливо для офлайн-навчання;

– клієнт-серверна архітектура з підтримкою офлайн-режиму гарантує безперервність навчального процесу та збереження прогресу користувача;

– можливість створення персоналізованих наборів та перегляд аналітики дозволяє адаптувати навчальний процес під індивідуальні потреби користувача;

– подальший розвиток системи передбачає розширення бази кандзі, вдосконалення архітектури CNN для підвищення точності розпізнавання та інтеграцію генеративних моделей для автоматичного створення мнемонічних історій.

Список використаних джерел

1. Heisig J. W. Remembering the Kanji 1: A Complete Course on How Not to Forget the Meaning and Writing of Japanese Characters. 6th ed. Honolulu : University of Hawai‘i Press, 2011. 496 p.
2. Ebbinghaus H. Memory: A Contribution to Experimental Psychology. New York : Teachers College, Columbia University, 1913. 123 p.
3. Woźniak P. A., Gorzelańczyk E. J. Optimization of repetition spacing in the practice of learning. *Acta Neurobiologiae Experimentalis*. 1994. Vol. 54. P. 59–62.
4. LeCun Y., Bengio Y., Hinton G. Deep learning. Nature. 2015. Vol. 521, No. 7553. P. 436–444. DOI: 10.1038/nature14539
5. Paivio A. Mental Representations: A Dual Coding Approach. New York : Oxford University Press, 1986. 322 p.
6. KANJIDIC2 Home Page. Electronic Dictionary Research and Development Group. URL: http://www.edrdg.org/kanjidic/kanjd2index_legacy.html (дата звернення: 18.10.2026).
7. KanjiVG: Kanji vector graphics. GitHub. URL: https://github.com/KanjiVG/kanjivg (дата звернення: 18.10.2026).
8. Левківський В., Марчук Д., Кравченко С., Павленко О. Розроблення мобільного додатку для вивчення іноземних мов з використанням машинного навчання. *Інформаційні технології і засоби навчання*. 2024. Том 102, № 4. С. 163–175. DOI: 10.33407/itlt.v102i4.5729
9. Kukulska-Hulme A., Shield L. An overview of mobile assisted language learning: From content delivery to supported collaboration and interaction. *ReCALL*. 2008. Vol. 20, No. 3. P. 271–289.
10. Mayer R. E. Multimedia Learning. 3rd ed. Cambridge : Cambridge University Press, 2020. 450 p.
11. Python Documentation. URL: https://docs.python.org/3/ (дата звернення: 18.10.2026).

УДК 004.4

Ігор Кандиба
PhD, доцент (б.в.з.) кафедри
інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ВИЗНАЧЕННЯ ТОНАЛЬНОСТІ КОМЕНТАРІВ ДО МЕДІАКОНТЕНТУ ВСТУПНОЇ КАМПАНІЇ ЗАКЛАДУ ВИЩОЇ ОСВІТИ ЗАСОБАМИ PYTHON

Безпекова ситуація в Україні, проблеми з електропостачанням та освітні втрати під час навчання в школі призвели до відтоку абітурієнтів закладів вищої освіти за кордон, а це, в свою чергу, збільшило конкуренцію між різними спеціальностями під час проведення вступної кампанії [1]. Ці факти зумовлюють актуальність створення спеціалізованого інструментарію для аналізу коментарів в соціальних мережах, що дасть змогу визначати фактори, які впливають на вибір абітурієнтами закладу вищої освіти та спеціальності. Особливо важливим є аналіз згаданих коментарів під медіаконтентом, який є одним з основних засобів поширення реклами під час вступної кампанії [2].

Важливим кроком в процесі аналізу коментарів до мендіаконтенту вступної кампанії закладу вищої освіти є визначення їх тональності та емоційного забарвлення. Проте в сучасному світі кожен заклад освіти використовує сторінки в різних соціальних мережах для поширення інформації, що ускладнює аналіз коментарів. Спростити цей процес можна шляхом використання мови програмування Python.

Мова програмування Python підтримує підключення сторонніх бібліотек, що спрощує розробку програмного забезпечення для вирішення різнорідних задач: розпізнавання образів, класифікація об'єктів на зображенні, створення та інтеграція моделей штучного інтелекту (ШІ), обробка природної мови тощо. У контексті аналізу коментарів в соціальних мережах особливо актуальним є напрям обробки природної мови. Python підтримує кілька бібліотек для обробки природної мови: NLTK, spaCy та pymorphy3. Ці бібліотеки можуть використовуватися для токенізації, лематизації, визначення частин мови, видалення ключових слів та інших операцій, необхідних для підготовки текстових даних до подальшого аналізу. Варто

зазначити, що згадані бібліотеки мають суттєві обмеження, зокрема недостатню підтримку української мови та складність реалізації визначення тональності тексту. Тому доцільніше використовувати інструментарій що базується на ШІ. Прикладами таких бібліотек є PyTorch та Transformers.

Бібліотека Transformers забезпечує взаємодію з великою кількістю попередньо навчених моделей архітектури Transformer і дає змогу застосовувати їх для задач класифікації тексту. Це невід'ємний крок для Визначення тональності, оскільки сучасні моделі можуть враховувати контекст слова та взаємозв'язки між словами, на відміну від простих підходів на основі словників. Цей інструмент має власний репозиторій моделей на платформі Hugging Face Hub, що значно спрощує пошук та інтеграцію відповідних моделей [3].

Модель ukr-detect/ukr-emotions-classifier дозволяє визначати тональність тексту написаного українською мовою. Завантажити цю модель на локальний комп'ютер можливо засобами Transformers, для цього використовується метод AutoTokenizer.from_pretrained( ). За наявності підключення до мережі Інтернет інтерпретатор Python відображатиме прогрес завантаження моделі (рис. 1).

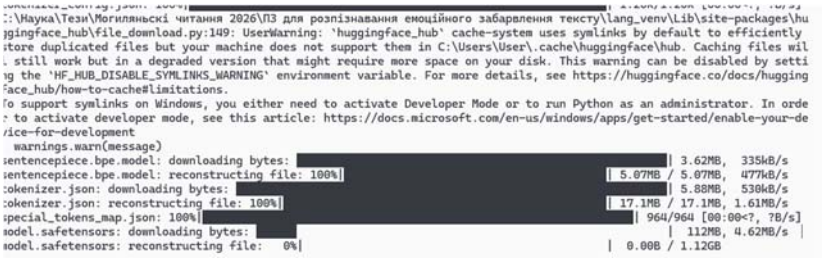


Рисунок 1. – Завантаження моделі
ukr-detect/ukr-emotions-classifier

Коректне використання інструментарію ШІ для визначення емоційного забарвлення тексту вимагає чіткого визначення порогових значень (thresholds). Порогові значення визначають межі прийняття рішень на основі оцінок, сформованих інструментарієм ШІ. Для моделей, вихідні оцінки яких належать до діапазону від 0 до 1, порогове значення визначає мінімальну оцінку, за якої відповідний клас вважається виявленим. Інструментарій Transformer дозволяє встановити значення порогові значення за допомогою вбудованої колекції Python – словника.

Після визначення порогових значень необхідно розділити текст на токени, які можуть бути оброблені моделлю ukr-detect/ukr-emotions-classifier. Цю обробку можна виконати за допомогою методу tokenizer( ), що реалізує розділення тексту на токени та перетворення їх у тензори. Тензор (tensor) представляє собою багатовимірну структуру даних, яка використовується для зберігання числових значень і виконання математичних операцій над ними.

Отримавши тензори для подальшої обробки необхідно використати завантажену модель:

```
        enc = tokenizer(texts, padding=True, truncation=True, return_tensors="pt")
        with torch.no_grad():
            outputs = model(**enc)
            logits = outputs.logits
            probs = torch.sigmoid(logits)
        id2label = model.config.id2label
        label_order = [id2label[i] for i in range(len(id2label))]

        predictions = []
        for prob_row in probs:
            single_pred = [
                label for label, prob in zip(label_order, prob_row.tolist())
                if prob > thresholds[label]
            ]
            predictions.append(single_pred)
        return predictions
```

Для перевірки розпізнавання емоцій було використано 8 коментарів, що містять різні емоції. В результаті виявлено проблему з розпізнаванням саркастичних висловлювань (рис. 2).

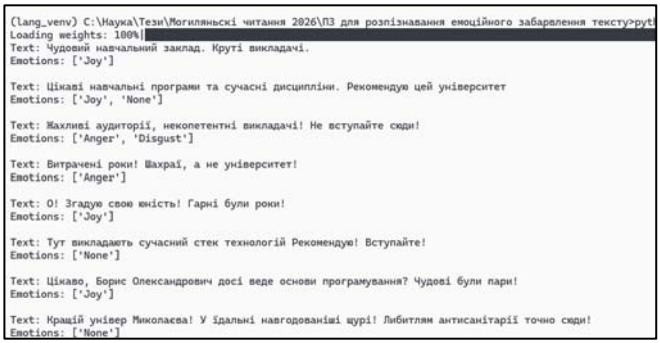


Рисунок 2. – Результат визначення тональності коментарів

Отримати поточні коментарі до медіаконтенту в соціальних мережах можливо за допомогою сторонніх бібліотек. Наприклад, для отримання даних з Telegram доступна бібліотека Telethon. Ця бібліотека дозволяє за допомогою API ключу керувати спеціалізованими ботом. Telegram зв’язує кожен пост публічного каналу з конкретним повідомленням у спеціальній пов'язаній (створеній) групі для обговорень, тому завантаження коментарів виконується через звернення до цієї групи за допомогою спеціального системного запиту: GetDiscussionMessage, який за ID поста знаходить головну гілку та зчитує всі повідомлення під нею.

Отже, визначення тональності коментарів до медіаконтенту вступної кампанії закладу вищої освіти може бути реалізоване засобами Python з використанням додаткових бібліотек. Засоби ШІ для обробки української мови розпізнають тональність коментарів, але виникають проблеми з розпізнаванням коментарів які містять сарказм та граматичні помилки. В майбутніх дослідження для виправлення цих недоліків планується застосування великих мовних моделей.

Список використаних джерел

1. Боровльова С. Ю., Гончаров Д. С., Горбань Г. В. Кандиба І. О. Виявлення освітніх втрат здобувачів вищої освіти засобами мови програмування Python. *Вчені записки ТНУ імені В. І. Вернадського. Серія : Технічні науки.* Том 36(75), № 1. С. 23–30.

2. Кандиба І. О., Фісун М. Т., Горбань Г. В. Генерація сценаріїв вступної кампанії закладу вищої освіти на основі когнітивної карти та предметно-орієнтованої мови програмування. *Вчені записки Таврійського національного університету імені В. І. Вернадського. Серія : Технічні науки.* Том 3, № 32. С. 96–104.

3. Osborne C., Ding J., Kirk H. R. The AI community building the future? A quantitative analysis of development activity on Hugging Face Hub. *Journal of Computational Social Science.* 2024. Vol. 7, Issue 2. P. 2067–2105.

УДК 378.147.88:004.9

Катерина Кірей
кандидат педагогічних наук, доцентка,
доцентка кафедри інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

МОДЕЛЮВАННЯ ЦИФРОВОГО ПРОФІЛЮ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ НА ОСНОВІ ІНТЕГРОВАНИХ ОСВІТНІХ ДАНИХ

Цифровізація вищої освіти супроводжується накопиченням значних обсягів даних про навчальну, адміністративну та активність здобувачів у соціальних мережах. Сучасні заклади вищої освіти (ЗВО) здебільшого функціонують в умовах фрагментації інформаційних ресурсів або «ізольованого» («силосного») зберігання даних. Адміністративна інформація зберігається в деканатських системах (ERP/CRM), аналітика навчальної поведінки – у системах управління навчанням (Moodle, Google Classroom тощо), а дані про додаткову залученість – у бібліотечних реєстрах, сервісах вебінарів та освітніх порталах. За браком єдиної концепції агрегації цих даних унеможливлюється проведення комплексного аналізу та впровадження прогностичних (предиктивних) моделей для розв'язання завдань освітньої аналітики.

У даній роботі розглянуто підходи до формування цифрового профілю здобувача вищої освіти на основі інтеграції різнотипних джерел освітніх даних, систематизовано основні вектори цифрового сліду здобувачів у середовищі ЗВО (системи керування навчанням (LMS), адміністративні системи, цифрові інструменти та зворотний зв'язок). Ґрунтуючись на цьому запропоновано концептуальну модель профілю для предиктивного аналізу успішності та персоналізації освітніх траєкторій.

Цифрова трансформація ЗВО передбачає перехід від простої фіксації успішності до комплексного аналізу навчального середовища та прийняття управлінських рішень на основі даних [1, 2]. Побудова цілісного цифрового профілю здобувача потребує систематизації джерел первинних даних за чотирма основними категоріями: навчальні платформи, адміністративні системи ЗВО, супутні цифрові інструменти, суб'єктивний зворотний зв'язок.

Дослідження джерел первинних даних дало змогу визначити їхню роль у навчальному процесі. Так навчальні платформи відображають процесуальну (поведінкову) складову навчання. До ключових метрик належать: частота та тривалість сесій, темп проходження матеріалів, хронологія подання завдань, результати поточного оцінювання та рівень комунікативної активності на форумах [1]. Адміністративні системи ЗВО містять базовий контекст та підсумкові показники. До них належать підсумкова академічна успішність (підсумкові бали, сесійні показники), демографічні характеристики, історія відвідуваності очних / онлайн-занять, статус виконання фінансових зобов'язань [1, 2]. Супутні цифрові інструменти характеризують ширшу залученість здобувача до освітнього середовища. Дані охоплюють використання електронних бібліотечних каталогів, активність у вебінарах, взаємодію з інституційними чат-ботами, а також метрики проходження інтерактивних модулів на основі віртуальної та доповненої реальності (AR/VR). Суб'єктивний зворотний зв'язок охоплює результати оцінювання якості викладання, самооцінку задоволеності освітніми компонентами та відгуки здобувачів.

Далі пропонується концептуальна модель цифрового профілю здобувача освіти. Цифровий профіль розглядається як багатовимірний динамічний масив даних, що інтегрує вказані джерела за допомогою єдиних ідентифікаторів користувача. Структура профілю охоплює такі основні блоки: академічний, поведінковий (цифровий слід), блок залученості, психолого-емоційний та рефлексивний блоки.

Академічний блок відображає підсумкові та проміжні результати освітньої діяльності, фіксуючи рівень засвоєння компетентностей та прогрес у виконанні навчального плану. Вхідними метриками та показниками є:

– абсолютна успішність (поточні бали, результати підсумкових контролів, модульних тестувань та іспитів (за 100-бальною та національною шкалами));

– відносна успішність або рейтинг (академічний рейтинг здобувача серед здобувачів курсу/спеціальності, відхилення від середнього балу групи (*Z*-оцінка));

– динаміка ЄКТС (кількість успішно зарахованих кредитів відносно загального обсягу за навчальним планом, частка прострочених або незарахованих кредитів);

– стабільність оцінювання (варіативність оцінок між різними дисциплінами, що може характеризуватися, зокрема, дисперсією, наприклад, гуманітарного та технічного блоків).

Джерелами даних є адміністративна система ЗВО (ERP/деканат), електронні відомості та відповідні АСУ.

Призначенням академічного блоку є формування базової академічної історії здобувача, розрахунок стипендіальних рейтингів та контроль виконання освітньої програми.

Поведінковий блок (цифровий слід) фіксує безпосередню навчальну поведінку здобувача в електронному освітньому середовищі, відображаючи його фактичний рівень дисциплінованості та самоорганізації. Вхідними метриками та показниками є:

– патерни роботи з контентом (час перегляду лекцій, рівень опрацювання контенту навчальних матеріалів (PDF, відео, презентації), тривалість перебування на сторінках курсів);

– часовий профіль активності (розподіл дій протягом доби (денна, нічна робота) та днів тижня);

– регулярність та дотримання дедлайнів (коефіцієнт здавання завдань напередодні дедлайну, частота запізнень зі здаванням робіт, інтервали між входами на платформу);

– комунікативна поведінка (кількість повідомлень у навчальних форумах, взаємодія з іншими здобувачами та викладачами під час групових обговорень).

Джерелами даних є логи та події LMS (Moodle, Google Classroom, Canvas тощо), отримані через протоколи xAPI (Tin Can) або Caliper Analytics.

Призначенням поведінкового блоку є раннє виявлення здобувачів, які втрачають темп навчання або схильні до прокрастинації, ще до початку екзаменаційної сесії.

Блок залученості вимірює рівень інтеграції здобувача в загальноуніверситетське цифрове та інфраструктурне середовище поза межами обов'язкових курсів LMS. Вхідними метриками та показниками є:

– бібліотечна та наукова активність (частота звернень до електронного каталогу, завантаження фахових публікацій із репозитарію, доступ до передплачених наукометричних баз (Scopus, Web of Science));

– комунікація та сервіси (кількість запитів до інституційних чат-ботів, відвідуваність освітнього порталу, користування сервісами підтримки);

– інноваційна залученість (час, проведений у спеціалізованих AR/VR-лабораторіях, симуляторах чи інтерактивних тренажерах).

Джерелами даних є автоматизовані бібліотечні інформаційні системи (АБІС), логи автентифікації освітніх порталів та системи корпоративного зв'язку.

Призначенням блоку залученості є розрахунок інтегрального індексу залученості, оцінка рівня залученості до університетського цифрового середовища та відчуття належності до університетської спільноти.

Психолого-емоційний та рефлексивний блок характеризує суб'єктивне сприйняття навчального процесу здобувачем, його психологічний комфорт, мотивацію та задоволеність якістю освітніх послуг. Вхідними метриками та показниками є:

– індекс задоволеності (кількісна оцінка окремих освітніх компонентів, курсів, викладачів та матеріально-технічного забезпечення за результатами систематичних опитувань);

– емоційний фон та зворотний зв'язок (аналіз текстових коментарів і відгуків за тональністю для виявлення слів із негативним або занепокоєним забарвленням);

– самооцінка та рефлексія (результати анкетування щодо рівня стресу, академічного вигорання, власної оцінки важкості засвоєння окремих тем);

– динаміка запитів на підтримку (частота звернень до психологічної служби ЗВО, кураторів груп).

Джерелами даних є системи внутрішнього анкетування, зворотний зв'язок у чат-ботах, опитувальники якості освіти та форми самооцінювання.

Призначенням психолого-емоційного та рефлексивного блоку є виявлення чинників академічного стресу, коригування навантаження та змісту навчальних дисциплін, а також запобігання відрахуванню за власним бажанням через емоційне вигорання.

Інтегрований цифровий профіль є основою для реалізації інструментів Data-Driven Education [1, 2]. До них належать системи раннього попередження – автоматичне виявлення здобувачів «групи ризику» (академічна заборгованість, загроза відрахування) на основі аналізу аномалій у поведінковому профілі (наприклад, суттєве зниження частоти входу в LMS у поєднанні з браком взаємодії з бібліотечними ресурсами); персоналізація навчальних траєкторій, що полягає в адаптації освітнього вмісту до індивідуального темпу здобувача та наданні автоматизованих рекомендацій щодо опрацювання додаткового матеріалу; оптимізація освітнього менеджменту за допомогою прийняття обґрунтованих управлінських

рішень щодо модернізації навчальних планів на основі об’єктивних аналітичних зрізів.

Треба зазначити, що реалізація цієї моделі потребує розв’язання низки завдань, зокрема, гарантування приватності даних відповідно до норм чинного законодавства, уніфікації форматів передавання даних (використання специфікацій xAPI та IMS LTI) та запобігання алгоритмічній упередженості під час оцінювання потенціалу здобувачів.

Отже, запропоноване моделювання цифрового профілю здобувача вищої освіти на основі інтеграції даних із системи керування навчанням, адміністративних систем та цифрових сервісів дає змогу перейти від статичної оцінки результатів навчання до безперервного аналізу освітнього процесу. Це створює підґрунтя для підвищення якості вищої освіти завдяки своєчасному реагуванню на проблеми здобувачів та створенню умов для персоналізованого навчання.

Список використаних джерел

1. Gaftandzhieva S., Hussain S., Hilcenko S., Doneva R., & Boykova K. (2023). Data-driven Decision Making in Higher Education Institutions: State-of-play. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 14(6), 374–385. DOI: https://doi.org/10.14569/IJACSA.2023.0140642

2. Aliyeva D. (2026). Data-driven decision making in higher education institutions. *The Online Journal of New Horizons in Education*, 16(2). URL: https://www.tojned.net/journals/tojned/articles/v16i02/v16i02-04.pdf

УДК 004.8:004.41

Євгеній Кітань
здобувач другого (магістерського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Катерина Антіпова
PhD, доцентка (б. в. з.)
кафедри інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ВИКОРИСТАННЯ АГЕНТІВ ШТУЧНОГО ІНТЕЛЕКТУ В РОЗРОБЦІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ: ПРАВИЛА, ПРОМПТИ ТА ЗОВНІШНІ ІНСТРУМЕНТИ

Ще кілька років тому штучний інтелект у роботі програміста зводився до автодоповнення і чат-бота, якому можна було вставити фрагмент коду й спитати, що з ним не так. Зараз дедалі більше розробників працюють з агентами програмування, такими як Claude Code, OpenAI Codex або агентний режим GitHub Copilot [1, 2]. Агент сам читає файли проєкту, редагує код, запускає команди й тести, тому від його налаштування залежать і якість результату, і безпека репозиторію. Усталених правил такого налаштування поки немає: кожен інструмент має власні файли конфігурації, а практичні поради розпорошені по документації.

Мета даної роботи – узагальнити практичні правила налаштування агентів програмування, зокрема написання файлів правил і промптів, використання навичок, підключення зовнішніх інструментів через протокол MCP, розподіл роботи між субагентами та обмеження дій агенту за допомогою дозволів і хуків.

Агентом програмування називають велику мовну модель, якій дали інструменти. Отримавши завдання, агент сам вирішує, які файли відкрити і які команди виконати, дивиться на результат і робить наступний крок. Усе, що він знає про поточне завдання, міститься в його контекстному вікні: запит користувача, прочитані файли, вивід команд. Розмір вікна обмежений, і що більше в ньому зайвого, то гірше модель утримує в увазі важливе. Крім того, кожна сесія починається з чистого контексту, тому агент не пам’ятає, як у проєкті запускаються тести і яких домовленостей дотримується команда [1].

Цю інформацію агенту передає файл правил проєкту. У Claude Code він називається CLAUDE.md, багато інших агентів читають AGENTS.md [1, 2]. Файл лежить у корені репозиторію, завантажується на початку кожної сесії і зазвичай містить:

– команди збирання проєкту і запуску тестів;

– опис структури проєкту і стилю коду;

– дії, які агент не повинен виконувати без дозволу, наприклад зміну міграцій бази даних чи публікацію змін у віддалений репозиторій.

Писати правила треба конкретно. Інструкцію «пиши якісний код» агент не може ні виконати, ні перевірити, а інструкцію «перед кожним комітом запускай npm test і не роби коміт, якщо тести падають» він виконає. Довгий файл теж шкодить, бо серед загальних слів легко пропустити важливе. Крім проєктного файлу, розробник може мати особисті правила для всіх своїх проєктів, організація може задати спільні правила для всіх розробників, а частину правил можна прив'язати до певних типів файлів. Агент сприймає ці файли як контекст, а не як обов'язкову конфігурацію, тому іноді може відступити від написаного. Claude Code також веде автоматичну пам'ять: агент сам записує нотатки про виправлення і вподобання користувача, і вони теж завантажуються на початку сесії [1].

Ті самі вимоги стосуються промпту для конкретного завдання. Гарний промпт містить [1]:

– мету, завдання і причину, з якої завдання треба виконати;

– контекст, тобто де лежить потрібний код і що вже пробували;

– вимірювані обмеження, наприклад «якщо після трьох спроб тести не проходять, зупинись і опиши проблему»;

– спосіб перевірки результату: які тести запустити або яке значення має вийти.

Для складних змін корисно спершу попросити агента скласти план і нічого не змінювати, бо помилку в розумінні задачі простіше виправити до того, як агент переписав десяток файлів. Найкраще агенту вдаються завдання з результатом, який можна перевірити: виправити помилку, на яку є тест, перенести код на нову версію бібліотеки, написати тести до наявного модуля. Велике завдання краще розбити на кілька менших і перевіряти результат кожного кроку [1].

Для повторюваних процедур зручно використовувати навички (skills). Навичка зберігається як папка з файлом SKILL.md, де описано, як виконувати певну роботу, наприклад готувати реліз чи писати міграцію. Агент постійно бачить лише короткий опис навички, а повний текст завантажує, коли вирішить, що навичка потрібна. Опис

«допомагає з кодом» підходить до будь-якого запиту, тому агент або завантажуватиме таку навичку постійно, або не завантажить ніколи. Опис «використовуй, коли треба створити або змінити міграцію бази даних у Prisma» прямо каже, коли навичка потрібна [1].

Для того, щоб агент міг працювати не лише з файлами, до нього підключають MCP-сервери. Model Context Protocol (MCP) є відкритим протоколом на основі JSON-RPC, через який застосунок з мовною моделлю отримує доступ до зовнішніх інструментів і даних [3]. Сервер для GitHub, наприклад, дає агенту змогу читати задачі й pull request'и, сервер бази даних виконувати запити, а сервер браузера відкривати сторінки [1]. Протокол розрізняє три сторони [3]:

– хост, тобто застосунок, з яким працює користувач, наприклад агент програмування;

– клієнт усередині хоста, який підтримує з'єднання з одним сервером;

– сервер, який надає ресурси (дані для моделі), шаблони промптів та інструменти, тобто функції, які модель може викликати.

Підключати всі доступні сервери підряд не варто. Опис кожного інструмента займає місце в контексті моделі і витрачає токени в кожному запиті [4]. Чим більше схожих інструментів, тим частіше агент обирає не той або викликає інструмент там, де міг відповісти сам. До того ж інструмент MCP фактично означає виконання чужого коду, а сервер, що отримує вміст з інтернету, може передати агенту промпт-ін'єкцію, тобто приховану інструкцію в даних [1, 5]. Специфікація протоколу тому вимагає, щоб користувач явно погоджувався на виклик інструментів [3].

Велике завдання агент може частково передати субагентам. Субагента описують окремим файлом з назвою, описом, списком дозволених інструментів і власним системним промптом, а працює він в окремому контексті. Якщо треба з'ясувати, де в проєкті обробляються платежі, субагент може переглянути десятки файлів і повернути основному агенту короткий висновок. Задачу субагенту передають за його описом, тому опис має бути коротким і точним, а інструменти краще обмежувати: субагенту, який лише шукає по коду, не потрібні права на редагування файлів [1].

Обмежити дії агента можна кількома способами [1]:

– правилами allow, ask і deny у файлі settings.json, які перевіряються в порядку deny, ask, allow, тож заборону не можна скасувати дозволом;

– хуками PreToolUse, які запускаються перед кожним викликом інструмента і блокують його, якщо завершуються з кодом 2;

– вибором режиму роботи, причому режим без перевірки дозволів документація радить вмикати лише в ізольованому середовищі, наприклад у контейнері.

Хуки корисні й для звичайної автоматизації. Хук PostToolUse спрацьовує після виклику інструмента і може, наприклад, запускати форматер після кожного редагування файлу, тож стиль коду не залежить від того, чи згадав агент про відповідне правило [1].

Різниця між правилами і хуками принципова. Правило в CLAUDE.md агент може не виконати, наприклад якщо запит сформульовано незвично, а заборону в дозволах чи хук він обійти не може. Префіксні правила теж мають слабке місце: правило проти git push не спрацює на команду git -C . push, хоча вона робить те саме. Хук, який розбирає саму команду, такого недоліку не має. Так само варто закрити правилом deny читання файлів на кшталт .env з ключами доступу [1].

Навіть добре налаштований агент не звільняє розробника від перевірки. Зміни агента варто переглядати так само, як код колеги на рев'ю, а працювати йому краще в окремій гілці з невеликими комітами, щоб невдалу зміну було легко відкотити [1]. Якщо агент кілька разів повторює ту саму помилку, краще дописати відповідне правило в CLAUDE.md, тоді наступна сесія вже почнеться з цього знання.

Отже, результат роботи агента програмування залежить від того, як розробник його налаштував. За результатами розгляду можна зробити такі висновки:

– правила проєкту і промпти мають бути короткими, конкретними й містити вимірювані обмеження, інакше агент діє за загальними звичками;

– опис навички чи субагента визначає, чи агент узагалі ними скористається, тому його слід писати під конкретні сценарії;

– зовнішні інструменти через MCP варто підключати вибірково, бо кожен з них витрачає контекст і може стати джерелом промпт-ін'єкцій;

– дії, яких агент не повинен виконувати за жодних умов, слід закривати дозволами і хуками, бо правило в текстовому файлі він може не виконати;

– зміни агента потрібно перевіряти так само, як код колеги, а помилки, що повторюються, записувати в правила проєкту.

Список використаних джерел

1. Claude Code Docs. Anthropic. URL: https://code.claude.com/docs/en/overview (дата звернення: 22.10.2026).
2. AGENTS.md. URL: https://agents.md/ (дата звернення: 22.10.2026).
3. Model Context Protocol Specification. Version 2025-11-25. Model Context Protocol. URL: https://modelcontextprotocol.io/specification/2025-11-25 (дата звернення: 22.10.2026).
4. Jones A., Kelly C. Code execution with MCP: Building more efficient agents. Anthropic. 04.11.2025. URL: https://www.anthropic.com/engineering/code-execution-with-mcp (дата звернення: 22.10.2026).
5. LLM01:2025 Prompt Injection. OWASP Gen AI Security Project. URL: https://genai.owasp.org/llmrisk/llm01-prompt-injection/ (дата звернення: 22.10.2026).

УДК 004.8:004.738.5

Лада Ковшун
здобувачка другого (магістерського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Стефанія Бондаренко
викладачка кафедри інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ІНТЕГРАЦІЯ ШТУЧНОГО ІНТЕЛЕКТУ В ЧАТ ДОПОМОГИ КЛІЄНТАМ

У сучасних умовах високої конкуренції у сфері послуг та ритейлу ефективне обслуговування клієнтів потребує гнучких, адаптивних і технологічно розвинених підходів. Використання інтелектуальних чатів допомоги дозволяє забезпечити цілодобову підтримку, оптимізувати навантаження на операторів, автоматизувати обробку типових запитів та підвищити загальний рівень задоволеності користувачів.

У сучасних умовах цифрової трансформації суспільства та економіки актуальним є застосування інтелектуальних систем для автоматизації процесів клієнтської підтримки. Традиційні методи обслуговування через call-центри та класичні текстові форми дедалі частіше доповнюються або замінюються інструментами, що базуються на штучному інтелекті (надалі, ШІ), машинному навчанні та аналітиці великих даних [1]. Це дозволяє підвищити швидкість обробки звернень, оптимізувати розподіл ресурсів відділу підтримки та знизити операційні витрати [2].

Інтелектуальний чат допомоги передбачає використання програмних комплексів, здатних автоматично аналізувати зміст повідомлень клієнтів, розпізнавати їхні наміри, визначати тональність звернення та пропонувати оптимальні відповіді або рішення [3]. Такі системи поєднують інструменти Data Mining, обробки природної мови, генеративного штучного інтелекту та систем підтримки прийняття рішень [4].

Для сучасних компаній ключовим аспектом є поєднання оперативного обслуговування з глибинною аналітикою поведінки клієнтів. Інтелектуальні чати можуть автоматично аналізувати дані CRM, історію покупок, попередні звернення та відгуки клієнтів, виявляючи типові проблеми та зони для покращення продуктів чи послуг. На основі цих даних формується алгоритм дій – від надання миттєвої довідкової інформації до автоматичного оформлення повернення товару чи зміни параметрів послуги.

В даній роботі розглядається застосування інтелектуальних алгоритмів у сфері підтримки клієнтів із використанням платформи Microsoft Azure Bot Service [5], інтегрованої з діалоговими моделями OpenAI [6], системами аналітики Power BI та CRM-платформою Microsoft Dynamics 365. Це дозволяє створювати замкнене інформаційне середовище, у якому дані про звернення клієнтів постійно оновлюються та аналізуються у реальному часі.

Особливу увагу приділено реалізації алгоритмів прогнозування тематики звернень та розпізнавання намірів (Intent Recognition) на основі історичних даних. Модель машинного навчання оцінює вплив різних факторів (тип клієнта, історія замовлень, час доби, час перебування на сайті, поточні акційні пропозиції) на складність та характер запиту. На основі прогнозів система пропонує автоматичне вирішення проблеми або маршрутизацію діалогу на найбільш компетентного оператора [7].

Інтелектуальна система підтримки також інтегрує модулі розпізнавання емоційного стану клієнта шляхом аналізу тексту за

допомогою NLP-алгоритмів (Sentiment Analysis) [3]. Це дозволяє виявляти критичні або негативно налаштовані звернення ще до ескалації конфлікту та пріоритетно передавати їх досвідченим менеджерам (рис. 1 та рис. 2).

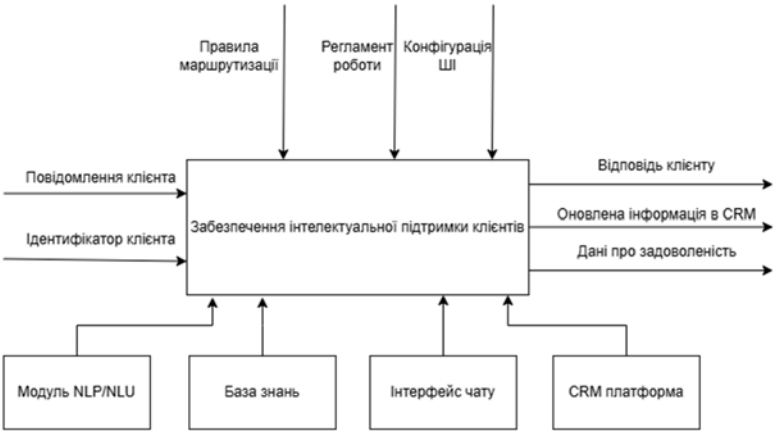


Рисунок 1. – IDEF0 структурного огляду системи: управління

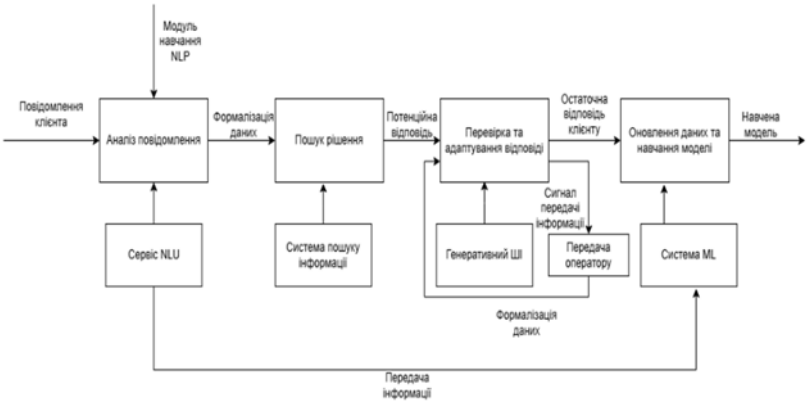


Рисунок 2. – Продовження IDEF0: основні механізми роботи

Застосування інтелектуальних технологій у чатах допомоги створює передумови для переходу до адаптивних систем обслуговування, які самостійно навчаються на основі успішно

вирішених діалогів. Такий підхід забезпечує зниження операційного навантаження на першу лінію підтримки, підвищення точності відповідей і гнучкість реагування на масові запити.

Виходячи з наведеного, можна сформулювати основні вимоги до інтелектуальної системи чату допомоги клієнтам:

– провести аналіз наявних методів та інструментів автоматизації клієнтської підтримки;

– розробити архітектуру інтелектуального чату з використанням ШІ-модулів розпізнавання мови та аналізу тональності;

– реалізувати механізми інтеграції з базами даних CRM/ERP та платформами аналітики;

– розробити інтерфейс візуалізації ключових показників ефективності (KPI) підтримки у реальному часі;

– протестувати систему на прикладі реального або симульованого потоку звернень клієнтів.

Додатковим аспектом розвитку інтелектуальних чатів підтримки є використання технологій предиктивної та прескриптивної аналітики, що дозволяють не лише відповідати на запитання, а й передбачати можливі проблеми клієнта та пропонувати найкращі дії для їх запобігання. Такі системи ґрунтуються на методах статистичного моделювання, глибинного навчання та обчислювального інтелекту, що забезпечують адаптивність системи до постійно змінних запитів ринку.

Впровадження алгоритмів прогнозування дозволяє визначати періоди пікового навантаження на чат, передбачати нестачу операторів і автоматично коригувати графіки чергувань. Алгоритми NLP використовуються для детального аналізу текстових звернень клієнтів, що допомагає швидко виявляти технічні збої у сервісі або покращувати якість обслуговування.

Інтелектуальна система чату допомоги також підтримує автоматичний контроль KPI – таких як швидкість першої відповіді, час вирішення проблеми, рівень задоволеності клієнтів (CSAT/NPS) та частка автоматично закритих звернень. На основі цих показників система може формувати рекомендації для керівництва, наприклад, щодо оновлення бази знань, запровадження нових самосервісів чи зміни скриптів спілкування.

Використання цифрових двійників процесів обслуговування дає змогу моделювати вплив реорганізаційних рішень – наприклад, змінення маршрутизації чатів, запуску нових каналів зв’язку або зміни регламенту відповідей. Це сприяє проактивному управлінню якістю

сервісу, коли рішення ухвалюються не реактивно, а на основі передбачення наслідків.

Людино-орієнтований підхід передбачає, що інтелектуальний чат не замінює повністю оператора підтримки, а виступає його асистентом, який аналізує контекст діалогу, підказує релевантні статті з бази знань та пропонує готовий шаблон відповіді. Це дозволяє зменшити рутину, підвищити точність рішень і зосередитися на вирішенні складних та нестандартних ситуацій [8].

Перспективним напрямом подальшого розвитку є інтеграція чатів допомоги з голосовими асистентами, мультиканальними платформами комунікацій, системами розпізнавання облич/голосу та предиктивною аналітикою клієнтського досвіду. Це забезпечить комплексне обслуговування за усіма точками контакту у єдиній інтелектуальній екосистемі.

Отже, інтеграція штучного інтелекту в чат допомоги клієнтам є ключовим інструментом цифрової трансформації, що підвищує ефективність підтримки, зменшує витрати часу та забезпечує конкурентні переваги. Застосування ШІ та аналітики даних у цій галузі дозволяє перейти до адаптивної, самонавчальної моделі обслуговування, здатної забезпечити стале зростання лояльності клієнтів і високу якість сервісу.

Сучасні дослідження показують, що поєднання інтелектуальних систем підтримки з агентно-орієнтованими технологіями дозволяє створювати децентралізовані моделі обслуговування, у яких кожен автономний ШІ-агент відповідає за власну категорію питань (наприклад, фінанси, логістика, технічні збої), але при цьому координується з іншими. Такий підхід підвищує гнучкість системи, мінімізує людський фактор і забезпечує швидке реагування на непередбачувані сплески звернень [9].

Перспективним напрямом розвитку чатів допомоги є впровадження цифрових двійників клієнтського шляху (Customer Journey Digital Twins) – віртуальних моделей, що відображають усі етапи взаємодії користувача з компанією. За допомогою таких моделей можна моделювати різні сценарії виникнення проблем, оцінювати вплив управлінських рішень, прогнозувати наслідки змін у роботі сервісів, що сприяє підвищенню точності оперативного реагування.

Крім того, важливу роль відіграє людино-орієнтований підхід [8], за якого система не витісняє людину, а виступає інтелектуальним ко-пілотом оператора, який надає підказки, аналізує альтернативи та пропонує обґрунтовані варіанти відповідей [1]. Така взаємодія створює синергію між людською емпатією і потужностями ШІ, що дозволяє

зберегти персоніфікований підхід та одночасно підвищити швидкість операційного обслуговування.

Впровадження інтелектуальних систем підтримки потребує розроблення нових методологій оцінки ефективності, які враховують не лише час відповіді та фінансові витрати, але й рівень автоматизації, адаптивність системи, точність розуміння природної мови та швидкість прийняття рішень.

Отже, інтелектуалізація клієнтської підтримки є ключовим етапом еволюції сучасного сервіс-менеджменту, що сприяє переходу від реактивних до проактивних і самонавчальних систем, здатних забезпечувати стале функціонування організацій у динамічному цифровому середовищі.

Підсумовуючи, можна зазначити, що інтеграція штучного інтелекту в чат допомоги клієнтам є стратегічним напрямом розвитку сучасного бізнесу, який поєднує технології генеративного ШІ, машинних моделей NLP, аналітики даних і цифрового моделювання. Завдяки цим технологіям клієнтський сервіс набуває нової якості – він стає проактивним, адаптивним і самонавчальним. Такий підхід дозволяє зменшити ризики втрати клієнтів, підвищити точність обслуговування, оптимізувати використання ресурсів відділу підтримки та скоротити витрати часу.

Впровадження інтелектуальних систем у сферу клієнтського сервісу сприятиме підвищенню конкурентоспроможності підприємств та розвитку інноваційної цифрової економіки загалом.

Список використаних джерел

1. Davenport T. H., Ronanki R. Artificial intelligence for the real world. *Harvard Business Review.* 2018. Vol. 96, No. 1. P. 108–116.

2. Chui M., Hazan E., Roberts R., et al. The economic potential of generative AI: the next productivity frontier. McKinsey & Company, 2023. URL: https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/the-economic-potential-of-generative-ai-the-next-productivity-frontier (дата звернення: 25.09.2026).

3. Jurafsky D., Martin J. H. Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition. 2nd ed. Upper Saddle River : Pearson Prentice Hall, 2009. 988 p.

4. ISO/IEC 22989:2022. Information technology – Artificial intelligence – Artificial intelligence concepts and terminology. Geneva : ISO, 2022. 60 p.

5. Azure AI Bot Service documentation. Microsoft Learn. URL: https://learn.microsoft.com/azure/bot-service/ (дата звернення: 25.09.2026).
6. OpenAI API documentation. OpenAI. URL: https://developers.openai.com/api/docs (дата звернення: 25.09.2026).
7. ISO/IEC 23053:2022. Framework for artificial intelligence (AI) systems using machine learning (ML). Geneva : ISO, 2022. 36 p.
8. ISO 9241-210:2019. Ergonomics of human-system interaction – Part 210: Human-centred design for interactive systems. Geneva : ISO, 2019. 33 p.
9. Russell S., Norvig P. Artificial intelligence: a modern approach. 4th ed. Harlow : Pearson Education Limited, 2022. 1166 p.

УДК 004.8

Микита Колесніков
здобувач третього (PhD) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Гліб Горбань
кандидат технічних наук, доцент,
доцент кафедри інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ОЦІНКА СТІЙКОСТІ КЛАСИФІКАТОРІВ ШТУЧНОГО КОНТЕНТУ ДО ЦІЛЕСПРЯМОВАНИХ ПРОГРАМНИХ ЗБУРЕНЬ

Класифікатори, що відрізняють реальні зображення та кадри відео від таких, що згенерованих штучним інтелектом, поступово стають частиною інфраструктури інформаційної безпеки. Їх застосовують для модерації контенту в соціальних мережах, фактчекінгу та перевірки медіадоказів. Однак висока точність таких систем на тестових вибірках ще не гарантує їх надійності в реальних умовах, де зловмисник свідомо намагається обійти перевірку. Глибокі нейронні мережі, на яких побудована більшість класифікаторів, вразливі до змагальних атак (Adversarial Attacks) – малих, непомітних для

людського ока змін вхідних даних, що змушують модель визнати згенерований контент реальним. Тому оцінка стійкості (робастності) класифікаторів до таких атак є необхідною умовою їх практичного застосування.

Формально задачу атаки можна описати так. Нехай f-класифікатор, який для вхідного зображення x повертає мітку $y \in$ {“реальне”, “згенероване”}. Змагальна атака полягає у пошуку такого збурення δ, що:

$$f\,(a+\delta) \neq y, \quad \|\delta\|_\infty \leq \varepsilon \tag{1}$$

де ε – бюджет збурення, що обмежує максимальну зміну значення кожного пікселя (для 8-бітних зображень зазвичай $\varepsilon \leq 8/255$).

Під програмними збуреннями розуміються атаки, що реалізуються виключно програмними засобами, шляхом модифікації цифрового файлу, на відміну від фізичних атак (друк і повторна зйомка, спеціальні маски тощо). Залежно від рівня доступу зловмисника до моделі їх можна поділити на три групи:

1) Атаки «білого ящика» (White-box), за яких зловмисник має повний доступ до архітектури та ваг моделі й обчислює збурення за градієнтом функції втрат (FGSM, PGD, атака Карліні – Вагнера (C&W));

2) Атаки «чорного ящика» (Black-box), що ґрунтуються на перенесенні змагальних прикладів, згенерованих на сурогатній моделі, або на ітеративних запитах до цільового класифікатора (Square Attack, HopSkipJump);

3) Трасформаційні атаки, які не потребують знань про модель і полягають у постобробці контенту: JPEG-стисненні, зміні розміру, розмитті, додаванні шуму, а також очищенні зображення дифузійною моделлю (Diffusion Purification).

Найпростішою атакою White-box є метод швидкого знаку градієнта FGSM (Fast Gradient Sign Method) [1], який формує змагальний приклад за один крок:

$$x_{\text{adv}} = x + \varepsilon \cdot \text{sign}\big(\nabla_x L(\theta, x, y)\big) \tag{2}$$

де L – функція втрат класифікатора; θ – його параметри; ∇_x – градієнт функції втрат за вхідним зображенням.

Значно сильнішою є ітеративна атака PGD (Projected Gradient Descent) [2], що повторює крок FGSM з меншим кроком α і після кожних ітерації повертає результат у допустиму область збурень:

$$x^{t+1} = \Pi_{B_\varepsilon(x)}\left(x^t + \alpha \cdot \text{sign}\left(\nabla_x L(\theta, x^t, y)\right)\right) \quad (3)$$

де П – операція проєкції на ε-окіл $B_\varepsilon(x)$ вихідного зображення; t-номер ітерації.

Принцип дії такої атаки схематично показано на рис. 1 – додавання до зображення збурення, яке людина практично не помічає, змінює рішення класифікатора на протилежне.

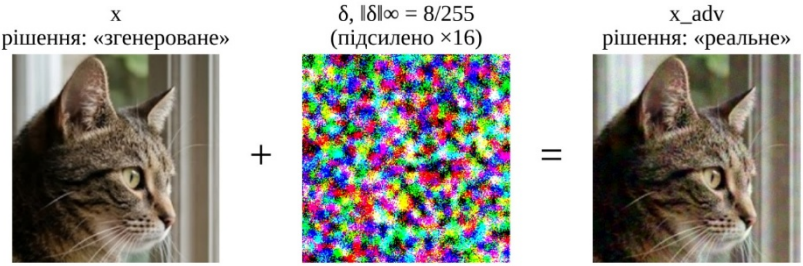


Рисунок 1. – Схема змагальної атаки на класифікатор штучного контенту (збурення δ підсилено для наочності)

Порівняльну характеристику груп програмних атак наведено в таблиці 1.

Таблиця 1. – Порівняльна характеристика груп програмних атак

Параметр	**White-box**	**Black-box**	**Трансформаційні**
Необхідний доступ до моделі	Повний (архітектура, ваги, градієнти)	Лише вихід моделі або сурогатна модель	Не потрібен
Помітність спотворень	Мінімальна	Низька або середня	Від низької до помітної
Обчилювальна складність	Від низької (FGSM) до високої (C&W)	Висока (тисячі запитів до моделі)	Низька

Параметр	White-box	Black-box	Трансформаційні
Ефективність проти класифікаторів	Дуже висока, близька до 100%	Середня або висока, залежить від переносимості	Залежить від стійкості моделі до постобробки
Основні обмеження для атакуючого	Потрібен доступ до моделі, чутливість до стиснення	Великий обсяг запитів, нижча успішність	Погіршення якості контенту

Практичну небезпеку таких атак підтверджують уже існуючі дослідження, наприклад в одному з таких детектор згенерованих зображень облич, що мав показник AUC 0,95, під дією різних атак втрачав здатність до розпізнавання – значення AUC знижувались до рівня, нижчого за випадкове вгадування [3] (рис. 2).

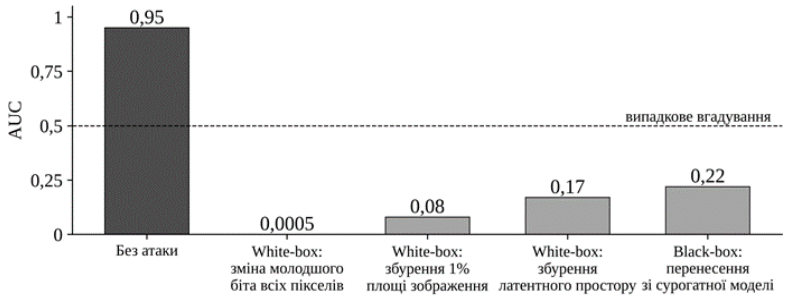


Рисунок 2. – Значення AUC детектора згенерованих зображень до та після атак [3]

Аналогічно, у дослідженні атак на детектори дипфейків XceptionNet та MesoNet, було досягнуто успішності понад 98% для нестиснених відео, хоча MJPEG-стиснення частково знижувало їх ефективність [4]. Автори обґрунтовують існування фундаментального компромісу між робастністю та надійністю класифікаторів штучних зображень [5].

Причина такої вразливості – у природі ознак, на які спираються класифікатори. Згенеровані зображення відрізняються від реальних переважно малоамплітудними високочастотними артефактами [6],

тому збурення співмірної величини здатне їх замаскувати. Не рятує й постобробка контенту в соціальних мережах, адже зловмисник може врахувати її під час генерування збурення за допомогою підходу Expectation over Transformation (EOT) [7]:

$$\delta^* = \arg\max_{\|\delta\|_\infty \leq \varepsilon} \mathrm{E}_{T\sim\mathcal{T}}[L(\theta, T(x+\delta), y)] \quad (4)$$

де $\mathcal{T}$ – розподіл очікуваних перетворень контенту (JPEG-стиснення, масштабування, обрізання); T – конкретне перетворення.

Таким чином, навіть класифікатор із високою точністю на тестовій вибірці можна обійти змагальною атакою, тому покладатись лише на нього під час перевірки контенту не варто. Змагальне навчання частково підвищує стійкість, але водночас погіршує результати на звичайних зображеннях. Через це – рішення детектора слід розглядати як одну з ознак, а не як остаточний висновок, а під час оцінювання таких систем, окрім точності, перевіряти і їх стійкість до атак.

Список використаних джерел

1. Goodfellow I. J., Shlens J., Szegedy C. Explaining and harnessing adversarial examples. 3rd International Conference on Learning Representations (ICLR 2015). 2015. URL: https://arxiv.org/abs/1412.6572 (дата звернення: 09.10.2026).
2. Towards deep learning models resistant to adversarial attacks / A. Madry, et al. 6th International Conference on Learning Representations (ICLR 2018). 2018. URL: https://arxiv.org/abs/1706.06083 (дата звернення: 19.10.2026).
3. Carlini N., Farid H. Evading deepfake-image detectors with white- and black-box attacks. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). 2020. P. 2804–2813. DOI: 10.1109/CVPRW50498.2020.00337
4. Adversarial deepfakes: evaluating vulnerability of deepfake detectors to adversarial examples / S. Hussain, et al. *2021* IEEE Winter Conference on Applications of Computer Vision (WACV). 2021. P. 3348–3357. URL: https://arxiv.org/abs/2002.12749 (дата звернення: 15.10.2026).
5. Robustness of AI-image detectors: fundamental limits and practical attacks / M. Saberi, et al. *12th International Conference on Learning Representations (ICLR 2024).* 2024. URL: https://arxiv.org/abs/2310.00076 (дата звернення: 16.10.2026).
6. Leveraging frequency analysis for deep fake image recognition / J. Frank et al. Proceedings of the 37th International Conference on Machine Learning (ICML 2020). *Proceedings of Machine Learning Research.* 2020.

Vol. 119. P. 3247–3258. URL: https://proceedings.mlr.press/v119/frank20a.html (дата звернення: 16.10.2026).

7. Synthesizing robust adversarial examples / A. Athalye, L. Engstrom, A. Ilyas, K. Kwok. Proceedings of the 35th International Conference on Machine Learning (ICML 2018). *Proceedings of Machine Learning Research.* 2018. Vol. 80. P. 284–293. URL: https://proceedings.mlr.press/v80/athalye18b.html (дата звернення: 16.10.2026).

УДК 004.738.5

Софія Крива
здобувачка другого (магістерського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євген Давиденко
кандидат технічних наук, доцент,
завідувач кафедри інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ІНТЕГРАЦІЯ ІНТЕРАКТИВНИХ ВЕБСЕРВІСІВ У ВИКЛАДАННЯ ІНОЗЕМНИХ МОВ У ЗАКЛАДАХ ЗАГАЛЬНОЇ СЕРЕДНЬОЇ ОСВІТИ

Сучасний етап трансформації національної системи освіти в контексті вимог Нової української школи (НУШ) та євроінтеграційних процесів вимагає докорінного оновлення методології та інструментарію викладання іноземних мов у середній школі [1, 2]. Англійська мова є універсальним засобом міжнародного спілкування та базовим інструментом у сфері інформаційних технологій [3]. З погляду програмної інженерії та комп'ютерних наук, ключовим завданням постає проєктування, розробка та практична інтеграція вебплатформ і мобільних застосунків, які забезпечують високу доступність, адаптивність, інтерактивність та кросплатформність освітнього процесу [1, 4, 5].

Застосування комп'ютерного мовного навчання (Computer-Assisted Language Learning, CALL) та сучасних засобів інформаційно-

комунікаційних технологій (ІКТ) дозволяє трансформувати здобувачів освіти з пасивних приймачів інформації на активних дослідників мовного середовища, підвищуючи їхню внутрішню мотивацію та самостійність [1]. Це забезпечує зацікавленість освітнім процесом за межами шкільного кабінету, перетворюючи вивчення мови на простір для творчості.

Згідно з дослідженнями американського психолога Р. Блейка, ефективність навчання залежить від відповідності педагогічним цілям засвоєння другої мови (Second Language Acquisition, SLA) та рівня підготовки викладача [1]. Водночас теоретик в галузі освіти М. Пренскі наголошує на появі «цифрових аборигенів» (Digital Natives) – покоління учнів, яке вимагає візуалізованого, динамічного та гейміфікованого навчального контенту [6].

При проєктуванні та програмній реалізації вебмодулів освітніх платформ для учнів середньої школи важливо дотримуватися таких вимог:

1) **Інтерактивність та гейміфікація UI/UX**, що полягає у впровадженні ігрових механік (нарахування балів, рейтинги, візуалізація прогресу) – стимулює пізнавальний інтерес та емоційну залученість учнів середнього шкільного віку [4, 6, 7];

2) **Адаптивність та персоналізація** через використання алгоритмів диференційованого оцінювання, що дає змогу автоматично формувати індивідуальні набори завдань відповідно до поточного рівня знань здобувача освіти [4, 5];

3) **Кросплатформність та доступність** завдяки забезпеченню стабільного функціонування веборієнтованих сервісів та мобільних застосунків на будь-яких пристроях (смартфони, планшети, ПК) [1, 7];

4) **Автоматизація контролю знань та аналітики** завдяки інтеграції програмних модулів для швидкої обробки результатів тестування та відображення аналітики успішності в режимі реального часу для вчителя та учня [5, 7].

У табл. 1 наведено порівняльну характеристику популярних інтерактивних цифрових інструментів, що застосовуються для формування іншомовної компетентності.

Таблиця 1. – Порівняння популярних інтерактивних цифрових інструментів

Назва інструменту, платформи	**Основні функціональні можливості**	**Переваги та дидактичний потенціал**
Kahoot!	Створення інтерактивних вікторин, опитувань, тестів та змагальних ігор у режимі реального часу [8]	Доступний інтерфейс, швидкий зворотний зв'язок, високий рівень гейміфікації та змагального ефекту [9]
Quizlet	Генерація інтерактивних карт (flash-cards), модулів тренування вимови, аудіосупроводу та різних ігрових режимів («Match», «Gravity»)	Ефективність для засвоєння та довготривалого запам'ятовування лексичних одиниць; можливість створення, контролю та поширення персоналізованих наборів
Classtime	Розширені типи запитань, інтеграція графіків, налаштування часових лімітів та автоматична перевірка відповідей	Гнучке налаштування навчальних сесій, підтримка рандомізації питань, розширена аналітика для викладача
MoDic, Words, Memrise	Створення власних електронних словникових карток, тренажери написання, озвучення диктором та автоматичні повторення [10]	Індивідуалізація засвоєння лексики, адаптація під складні для учня слова, робота в офлайн-режимі
ШІ-сервіси (ChatGPT, Duolingo, чат-боти)	Персоналізований аналіз мовлення, автоматичні підказки й виправлення граматичних помилок, інтерактивне діалогове спілкування	Якісний зворотний зв'язок, адаптація рівня складності складних граматичних структур, формування навичок говоріння

Серед аналізованих освітніх інструментів найновішим напрямом є сервіси на основі штучного інтелекту та генеративних мовних моделей (зокрема ChatGPT). Їх масове впровадження в освіту розпочалося наприкінці 2022 року. Якщо ж розглядати виключно спеціалізовані вебплатформи інтерактивного тестування – найновішою є Classtime, заснована 2016 року. Програмна реалізація таких систем базується на нейромережевій архітектурі Transformer із використанням мови програмування Python та фреймворків глибокого навчання (PyTorch, TensorFlow). Для забезпечення обробки великих масивів текстових даних і функціонування в режимі реального часу застосовуються високопродуктивні паралельні обчислення на графічних процесорах (GPU) за допомогою технології NVIDIA CUDA. Адаптація та персоналізація відповідей моделі під освітні завдання досягається завдяки методам тонкого налаштування (Fine-Tuning) та алгоритмам навчання з підкріпленням на основі зворотного зв'язку від людини (Reinforcement Learning from Human Feedback, RLHF), що дозволяє здійснювати точний аналіз мовних структур, граматичну корекцію та ведення інтерактивного навчального діалогу [11].

Важливим напрямом є міжпредметна інтеграція інформатики та іноземної мови. Наприклад, під час вивчення теми «Опрацювання текстових даних» у 5-6-х класах учні опановують англомовну термінологію (*text formatting, word processor, text insertion, copy and paste*), що одночасно розвиває цифрову й лінгвістичну компетентності [3].

Дж. Хедфілд класифікує гейміфікацію та ігрові технології наступним чином:

- ігри на відгадування;
- рольові ігри;
- настільні та кінетичні ігри.

Це сприяє зняттю мовного бар'єру та розвитку комунікативної компетентності в говорінні й аудіюванні [10, 12]. До того ж, використання графічних органайзерів та візуальних засобів забезпечує образне сприйняття та посилює запам'ятовування лексичного матеріалу до 65% при поєднанні візуального й аудіального каналів сприйняття [9].

Ефективність цифрових інструментів досить висока, проте їх впровадження супроводжується низкою викликів. До них входять цифровий бар'єр та технічне забезпечення, недостатня готовність педагогічного корпусу та ризик відволікання учнів, а також недостовірність інформації [1, 4].

Подальші дослідження у сфері комп'ютерних наук дають можливість більш широкої розробки мобільних адаптивних систем аналізу усного мовлення із застосуванням нейромережевих моделей та штучного інтелекту.

Список використаних джерел

1. Фодор К. Й. Використання інформаційних технологій у процесі викладання англійської мови у середній загальноосвітній школі. *Науковий вісник Мукачівського державного університету. Серія : Педагогіка та психологія.* 2018. Вип. 2(8). С. 167–169.
2. Довгань О., Матвєєва С. Цифрові технології в лінгвістиці: від мовних корпусів до штучного інтелекту : підручник. Київ : УДУ імені Михайла Драгоманова, 2025. 226 с.
3. Скурська Л. М. Загальні аспекти міжпредметної інтеграції навчання інформатики та англійської мови у закладах загальної середньої освіти. *Матеріали Всеукраїнської науково-практичної Internet-конференції «Автоматизація та комп'ютерно-інтегровані технології у виробництві та освіті: стан, досягнення, перспективи розвитку»*. Черкаси, 2024. С. 249–250.
4. Бокоч Т. Впровадження інформаційних технологій у навчання іноземних мов для фахівців-лінгвістів: сучасні підходи та виклики. *Актуальні питання гуманітарних наук*. 2024. Вип. 72, Т. 1. С. 191–195.
5. Гадайчук Н. М., Тульчак Л. В. Впровадження інформаційних технологій у дидактику іноземних мов у вищій школі. *Сучасні інформаційні технології та інноваційні методики навчання*. 2020. Вип. 66. С. 96–103.
6. Prensky M. Digital Natives, Digital Immigrants Part 2: Do They Really Think Differently? *On the Horizon*. 2001. Vol. 9, No. 6. P. 1–6.
7. Дячишина К. Використання інтерактивних тестів для підтримки змішаного навчання у процесі вивчення англійської мови: Kahoot і Quizlet. *Актуальні питання сучасної інформатики*. Житомир : Вид-во ЖДУ ім. І. Франка, 2024. Вип. 12. С. 65–68.
8. Шалацька Г. М. Використання сервісу Kahoot! у процесі вивчення іноземної мови за професійним спрямуванням. *Збірник наукових праць ЛНУ ім. Тараса Шевченка*. 2021. С. 112–115.
9. Quizlet та Kahoot у вивченні англійської мови. URL: https://duikt.edu.ua/ua/news-1-0-10690-quizlet-ta-kahoot-u-vivchenni-angliyskoi-movi?lang=ua&act=view&page=1&category=0&id=10690&sys_link=quizlet-ta-kahoot-u-vivchenni-angliyskoi-movi (дата звернення: 12.10.2026).
10. DeHaan, J. Teaching language and literacy (or anything) with games (or anything): A good way (The pedagogy of multiliteracies) simplified here

for teachers and students. *Ludic Language Pedagogy*. 2022. No. 4. P. 14–30. DOI: 10.55853/llp_v4Pg2

11.Ouyang L., Wu J., Jiang X. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*. 2022. Vol. 35. P. 27730–27744. DOI: 10.5555/3600270.3602281

12.Hadfield J. Elementary Communication Games 2.0. Pavilion Publishing and Media Ltd, 2024. 148 p.

УДК 004.94:631.5

Денис Поляк
здобувач другого (магістерського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Альона Швед
докторка технічних наук, професорка,
професорка кафедри інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

АВТОМАТИЗОВАНА ІНФОРМАЦІЙНА СИСТЕМА ОПЕРАТИВНОГО ПЛАНУВАННЯ ТА ЗБИРАННЯ ТОМАТІВ

Збирання томатів для промислової переробки відбувається протягом обмеженого сезону, а своєчасність робіт безпосередньо впливає на обсяг і якість сировини. Диспетчер підприємства має узгодити дозрівання десятків і сотень полів, обмежену продуктивність збиральних комплексів відділень та добову потребу переробного заводу. Ручне складання такого плану трудомістке, а будь-яка зміна вхідних даних, наприклад зупинка техніки чи коригування плану заводу, змушує переглядати розклад повністю. Науковці розглядають планування у томатній промисловості переважно на тактичному рівні: за допомогою лінійного та цілочислового програмування [1, 2], реактивного планування на основі імітаційного моделювання [3]. Роботи з календарного планування з обмеженими ресурсами [4] зосереджені на загальних задачах і не пов’язують збирання з допустимим вікном навколо прогнозної дати дозрівання. Тому

актуальним є створення спеціалізованої інформаційної системи, яка щоденно формує план збирання полів з урахуванням цих обмежень.

Мета даної роботи полягає у підвищенні ефективності оперативного планування збирання томатів за рахунок розробки автоматизованої інформаційної системи.

Постановка задачі. Планується збирання *F* полів, закріплених за *K* відділеннями, на горизонті з *n* діб. Для кожного поля *i* валовий збір визначається як

$$G_i = S_i \cdot Y_i, \tag{1}$$

де S_i – площа поля, га; Y_i – врожайність поля, т/га.

Прогнозна дата дозрівання R_i визначається як сума дати висадки та вегетаційного періоду. Допустиме збиральне вікно (ДЗВ) поля може бути задано інтервалом $[R_i - e;\ R_i + l]$, де e і l допустимі ранній і пізній зсуви збирання (за замовчуванням 1 і 7 діб).

Добова потужність відділення визначається як

$$C_k = (n_A \cdot q_A + n_B \cdot q_B) \cdot h, \tag{2}$$

де n_A, n_B – кількість комбайнів двох типів; q_A, q_B – годинна продуктивність (т/год); h – тривалість зміни.

Змінною розв'язку є обсяг збирання x_{it} поля *i* в добу *t*. Розклад є допустимим, якщо виконуються обмеження: з поля не збирається більше, ніж воно дає $(\sum x_{it} \le G_i)$; сумарний обсяг відділення за добу не перевищує C_k; збирання за добову потребу заводу не виходить за ліміт; збирання поза ДЗВ заборонене.

Критерієм є максимізація охоплення врожаю:

$$K = 100\,\% \cdot \frac{(\Sigma_i \Sigma_t\, x_{i,t})}{(\Sigma_i\, G_i)} \to \max, \tag{3}$$

План вважається прийнятним при $K \ge 99{,}9\%$.

Задача належить до класу *NP*-складних задач календарного планування з обмеженими ресурсами (*RCPSP*), тому для оперативного використання застосовано евристичний підхід [4, 5]. Кожне поле розглядається як активність з обсягом робіт G_i і часовим вікном ДЗВ, а добова потужність відділення як відновлюваний ресурс. Розклад будується жадібно по добах. На кожній добі активні поля впорядковуються за терміновістю

$$u_i = \left(\frac{r_i}{\max(1, DL_i - t)}\right) \cdot (1 + 0{,}5 \cdot s_i), \quad (4)$$

де r_i – залишок валу поля; DL_i – остання доба вікна; s_i – кількість діб поспіль, коли поле пропускалось.

Множник s_i запобігає «голодуванню» малих полів. Далі відділення розподіляє свій добовий ресурс між полями за спаданням u_i. Базовий план для заводу формується окремо без урахування ДЗВ, після чого оптимізується за вікном збирання. Якщо охоплення нижче порогу, система діагностує причину (поле за горизонтом планування, замале вікно, ресурсний конфлікт, недостатня загальна потужність), оцінює дефіцит комбайнів і автоматично підбирає мінімальні значення *e* та *l*, перебираючи пари, що мінімізують суму *e* + *l*.

Автоматизована інформаційна система оперативного планування та збирання томатів розроблена як настільний багатовкладковий застосунок на Python 3.13. Графічний інтерфейс побудовано на PySide6, обробка таблиць і обмін з Excel виконуються засобами pandas і openpyxl, графіки динаміки «дозріло – зібрано – залишок» реалізовані можливостями бібліотеки matplotlib. Дані зберігаються у файлах JSON, результати експортуються у XLSX і CSV, а тривалі обчислення, зокрема автоматичний підбір ДЗВ, виконуються у фонових потоках з можливістю скасування. Програмний застосунок складається з п’яти вкладок, що відповідають процесам функціональної моделі: підготовка вхідних даних, налаштування параметрів, формування базового плану, оптимізація за ДЗВ, контроль і подання результатів.

Адекватність та якість реалізованої математичної моделі оперативного планування та збирання томатів оцінено на контрольному прикладі з 105 полів у чотирьох відділеннях (валовий збір 287 898,1 т, дозрівання 2 серпня – 6 жовтня 2025 р.) та планом заводу на 66 діб. За сумарної добової потужності відділень 5 104 т система сформувала план, що охоплює 287 875,0 т, тобто $K \approx 99{,}992\%$ (з можливістю похибки в 0,5 т на поле). Це відповідає прийнятності плану за встановленим критерієм. Розроблена автоматизована інформаційна система додатково формує звіт про відхилення дат збирання від ДЗВ та попередження при недостатньому охопленні.

Висновки. Вирішено задачу оперативного планування збирання томатів у вигляді задачі календарного планування з обмеженими ресурсами та реалізовано автоматизовану інформаційну систему на основі жадібної евристики за терміновістю. Система дозволяє диспетчеру швидко отримувати та перераховувати план збирання при

зміні вхідних даних, контролювати охоплення врожаю та автоматично підбирати допустиме збиральне вікно. Подальший розвиток роботи передбачає врахування погодних факторів і технічних простоїв.

Список використаних джерел

1. Rocco C. D., Morabito R. Production and logistics planning in the tomato processing industry: A conceptual scheme and mathematical model. *Computers and Electronics in Agriculture*. 2016. Vol. 127. P. 763–774.
2. Alarcón Gerbier E. A., Gonzalez-Araya M. C., Rivera Moraga M. M. Supporting Harvest Planning Decisions in the Tomato Industry. Proceedings of the 6th International Conference on Operations Research and Enterprise Systems (ICORES 2017). 2017. P. 353–359.
3. Koulouris A., Kotelida I. Simulation-based reactive scheduling in tomato processing plant with raw material uncertainty. *Computer Aided Chemical Engineering*. 2011. Vol. 29. P. 1020–1024.
4. Hartmann S., Briskorn D. An updated survey of variants and extensions of the resource-constrained project scheduling problem. *European Journal of Operational Research*. 2022. Vol. 297, No. 1. P. 1–14.
5. Kolisch R. Serial and parallel resource-constrained project scheduling methods revisited: Theory and computation. *European Journal of Operational Research*. 1996. Vol. 90, No. 2. P. 320–333.

УДК 004.414.23

Михайло Прозапас
здобувач першого (бакалаврського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Микита Ковальчук
викладач кафедри інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ПРОЄКТУВАННЯ АРХІТЕКТУРИ МОБІЛЬНОЇ СИСТЕМИ ОБЛІКУ ПРОДУКТІВ І ПІДБОРУ РЕЦЕПТІВ

Раціональне використання продуктів харчування є важливим аспектом організації повсякденного життя. Відсутність систематизованого обліку харчових запасів ускладнює планування

покупок, спричиняє дублювання придбаних товарів та збільшує ймовірність псування продуктів. Одним із можливих рішень є розробка мобільної інформаційної системи, яка забезпечуватиме облік наявних продуктів, контроль термінів придатності, аналіз харчової цінності та автоматизований підбір рецептів.

Основою такої системи є правильно спроєктована програмна архітектура. Доцільно застосувати багатошаровий підхід, який передбачає розділення застосунку на рівень представлення, рівень бізнес-логіки та рівень доступу до даних. Відповідно до рекомендацій Android Developers [1], розмежування відповідальності між компонентами сприяє підвищенню тестування, надійності та супроводжуваності програмного забезпечення.

Рівень представлення забезпечує взаємодію користувача із системою через графічний інтерфейс. Бізнес-логіка реалізує обчислення харчової цінності, перевірку термінів придатності та алгоритми підбору рецептів. Рівень доступу до даних відповідає за збереження інформації про продукти, рецепти й налаштування користувача.

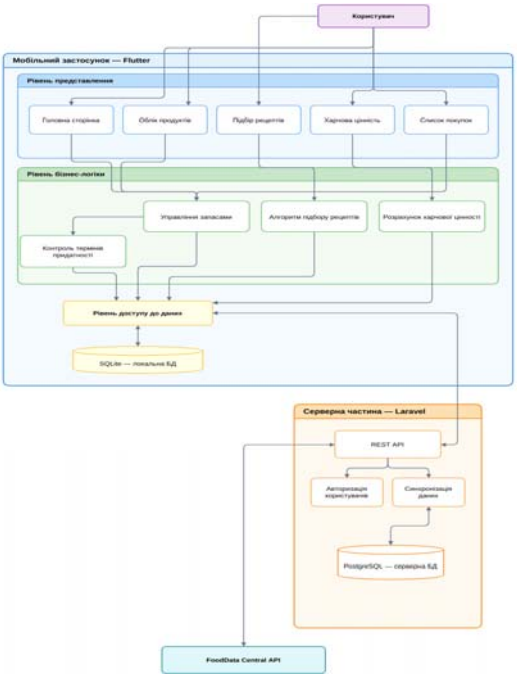


Рисунок 1. – Схема архітектури застосунку

Основним елементом системи є мобільний застосунок, розроблений із використанням фреймворку Flutter [2]. Його внутрішня структура складається з трьох основних рівнів:

1. Рівень представлення (UI) – відповідає за взаємодію користувача із застосунком. Містить головну сторінку, модулі обліку продуктів, підбору рецептів, перегляду харчової цінності та формування списку покупок;
2. Рівень бізнес-логіки – забезпечує виконання основних функцій системи: управління запасами продуктів, автоматизований підбір рецептів відповідно до наявних інгредієнтів, розрахунок харчової цінності та контроль термінів придатності
3. Рівень доступу до даних – забезпечує взаємодію бізнес-логіки з локальним сховищем і серверною частиною застосунку.

Для розробки мобільного застосунку управління харчовими запасами обрано клієнт-серверну архітектуру, яка поєднує можливості автономної роботи на пристрої користувача та надійну хмарну синхронізацію. Для локального зберігання інформації безпосередньо на смартфоні використовується SQLite [3], що дозволяє користувачеві переглядати та редагувати дані про продукти та рецепти навіть за відсутності підключення до Інтернету.

Централізована серверна частина розроблена за допомогою сучасного фреймворку Laravel [4], який відповідає за обробку вхідних запитів через REST API, безпечну авторизацію користувачів, а також за синхронізацію інформації між пристроями. Для централізованого зберігання даних та управління глобальною базою передбачено використання реляційної бази даних PostgreSQL [5].

Окрім внутрішніх компонентів, система взаємодіє із зовнішнім сервісом FoodData Central API, який надає детальну та актуальну інформацію про харчову цінність продуктів, зокрема їхню калорійність, а також кількість білків, жирів і вуглеводів [6].

Зв’язки між окремими блоками системи чітко відображають напрямки передачі даних та взаємодію її компонентів. Користувач виконує певну дію через графічний інтерфейс, після чого запит передається до відповідного модуля бізнес-логіки. За потреби відбувається звернення до локальної бази даних на пристрої або до серверного API. Отримані результати обробки повертаються назад до інтерфейсу та відображаються користувачеві.

Таким чином, запропонована архітектура забезпечує розмежування відповідальності між компонентами, підтримку автономної роботи, централізовану синхронізацію даних і можливість подальшого розширення функціональності. Це створює основу для розробки масштабованої та зручної мобільної системи управління харчовими запасами.

Список використаних джерел

1. Android Developers. Guide to app architecture. URL: https://developer.android.com/topic/architecture (дата звернення: 09.10.2026).
2. Flutter – Build apps for any screen. URL: https://flutter.dev/ (дата звернення: 10.10.2026).
3. SQLite. About SQLite. URL: https://sqlite.org/ (дата звернення: 14.10.2026).
4. Laravel. The clean stack for Artisans and agents. URL: https://laravel.com/ (дата звернення: 14.10.2026).
5. PostgreSQL: Documentation. PostgreSQL: The world's most advanced open source database. URL: https://www.postgresql.org/docs/ (дата звернення:15.10.2026).
6. Postman. API engineering for your agents. URL: https://www.postman.com/api-evangelist/agricultural-research-service-ars/collection/nex4lq6/food-data-central-api?sideView=agentMode (дата звернення:16.10.2026).

УДК 004.4

Данило Пройдисвіт
здобувач першого (бакалаврського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Ігор Кандиба
PhD, доцент (б.в.з.) кафедри
інженерія програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ІНСТРУМЕНТАРІЙ РОЗРОБКИ ВЕБЗАСТОСУНКУ АНАЛІЗУ ФІНАНСІВ

Головною метою у розробці сучасного фінансових програмного забезпечення для ведення фінансів є створення зручного багатокористувацького середовища для обліку коштів і надання можливості користувачеві аналізувати власні та групові грошові потоки в автоматичному режимі. Оскільки використання традиційних

способів ведення бюджету (як-от простих нотаток чи локальних електронних таблиць) не дає можливості синхронізувати спільні витрати між групою учасників та вчасно виявляти нетипові фінансові операції, доцільним є використання клієнт-серверних технологій, що поєднані із алгоритмами машинного навчання.

Впровадження і використання сучасних технологій веброзробки та машинного навчання та ШІ (штучного інтелекту) є дуже важливим у галузі фінансової індустрії [1]: до прикладів можна віднести новітні розробки різних платформ для групового бюджетування (як-от для сім'ї чи групи друзів), інструменти прогнозування грошових потоків та виявлення нетипових фінансових операцій на основі алгоритмів машинного навчання тощо.

Використання багатокористувацької платформи з інтеграцією ШІ надає великий функціонал користувачу, що надає наступні особливості: можливість одночасно працювати із приватним та груповими просторами, в режимі реального часу розділяти рахунки між учасниками груп, переглядати динамічні дашборди та покладатися на модель, що самотужки навчається на даних користувача, яка адаптується до звичок ведення фінансових операцій конкретної групи учасників.

Основою взаємодії користувача із системою є клієнтський вебзастосунок, що відповідає за введення фінансових операцій та візуалізацію даних (зведена аналітика). Реалізація цього вебзастосунку базується на використанні мови програмування TypeScript та фреймворку на основі React Next.js, що дозволяє створювати застосунки за технологією SPA (Single-Page Application) [2]. Завдяки такому підходу інтерфейс забезпечує неймовірно швидке перемикання між приватним та груповими фінансовими балансами, інтерактивне відображення рейтингів витрат за категоріями, різних діаграм постійних та змінних витрат, а також графіків щоденної динаміки навантаження на бюджет.

Також важливим є серверна частина вебзастосунку, яка є центральним компонентом обробки бізнес-логіки, валідації даних і керуванням безпекою. Реалізація цього компоненту базується на мові програмування Python, вебфреймворку FastAPI та ASGI-сервері Uvicorn. Безпечна автентифікація та захист сесій забезпечуються за допомогою пари токенів доступу / оновлення сесії за стандартом JWT [3], а розмежування повноважень всередині груп базується на рольовій моделі доступу, яка поділяє користувачів на головний акаунт, стандартного учасника та спостерігача.

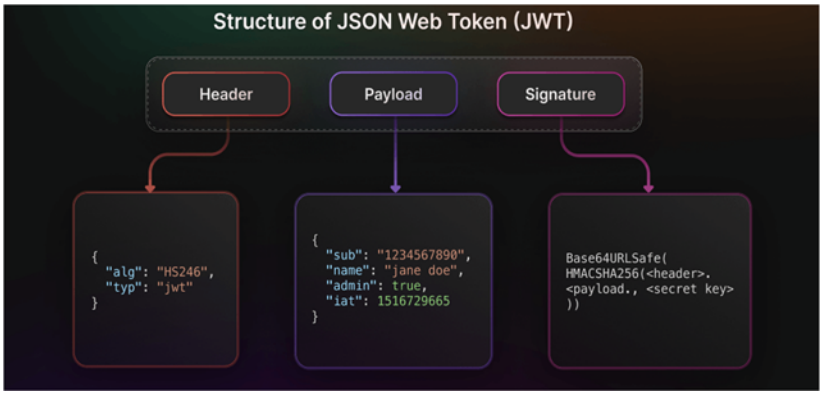


Рисунок 1. – Структура JWT

Аби мати змогу обробляти фінансові записи із використанням штучного інтелекту, у серверну частину інтегруються бібліотеки машинного навчання та аналізу даних, такі як Scikit-Learn, Pandas і NumPy. Оскільки для реалізації серверної частини вебзастосунку використано мову програмування Python, це дає змогу легко поєднувати REST API з моделями, що, у свою чергу, не вимагає переносити логіку на інші платформи. Модуль штучного інтелекту проводить початкову класифікацію транзакцій для нових користувачів і підтримує механізм активного навчання: якщо користувач часто змінює категорію вручну, система змінює індивідуальні ваги моделі або створює контекстне правило (тобто вона перенавчається на звичках користувача). Вона також поширює знайдені шаблони на спільні витрати групи. Алгоритми перевіряють історію операцій на повторюваність, щоб автоматично знаходити приховані підписки. Вони також рахують середнє значення і стандартне відхилення в кожній категорії, щоб вчасно помітити нетипові витрати.

Збереження і цілісність фінансових даних забезпечує реляційна система керування базами даних PostgreSQL. Взаємодія з нею на рівні програмного коду відбувається через об’єктно-реляційний прошарок ORM SQLAlchemy і схеми валідації Pydantic. Головні переваги PostgreSQL у контексті фінансового трекера: повна підтримка транзакційності за принципами ACID. Це не дає розсинхронізувати баланси при одночасних списаннях або переказах між рахунками. Система надійно працює зі складними реляційними зв’язками. Вона забезпечує високу швидкість агрегаційних запитів під час створення фінансових звітів.

Архітектура моделі даних і бізнес-логіка серверної частини базуються на моделях SQLAlchemy. Головними об'єктами системи є користувач і робочий простір. Вони з'єднуються через проміжну сутність члена групи із вказанням ролі учасника. До користувача або групи можна прив'язати об'єкти гаманців різних типів (як-от банківська картка і готівка). Є підтримка мультивалютності та лімітів. Фінансові операції виконуються через сутність транзакції, яка містить посилання на автора запису, рахунок, довідник категорій і прапорці постійної чи змінної витрати. Для механізму розподілу спільних чеків до транзакції додається сутність розбиття, що фіксує частку кожного учасника і дозволяє автоматично рахувати кількість боргів усередині групи. Персоналізовані ваги автоматичної категоризації зберігаються у таблиці правил ШІ.

Сучасні вебсервіси працюють у хмарній інфраструктурі, до прикладу, в Amazon Web Services (AWS). На етапі підготовки програмного продукту до запуску використовують технологію контейнеризації Docker та Docker Compose. З їх допомогою клієнтську частину, сервер, базу даних та сховище Redis поміщають в окремі контейнери разом з усіма потрібними залежностями. Проте під час роботи багатокористувацького застосунку виникають певні технічні проблеми, які стосуються продуктивності. Коли зростає кількість учасників груп, збільшується історія транзакцій і постійно виконуються складні запити до БД для побудови графіків та підрахунку статистичних відхилень ШІ, що підвищує навантаження на базу даних і збільшує час відгуку сервера. Якщо перенавчати ML-модель та виконувати інші складні операції одночасно із кожним запитом користувача, це може викликати затримки в роботі вебзастосунку. Тому для вирішення проблеми падіння швидкості під навантаженням у системі використовують додаткову оптимізацію – це кешування готових аналітичних вибірок і токенів сесій у швидкому сховищі Redis, застосування складених індексів у таблицях БД, серверна пагінація списків операцій і асинхронна обробка запитів.

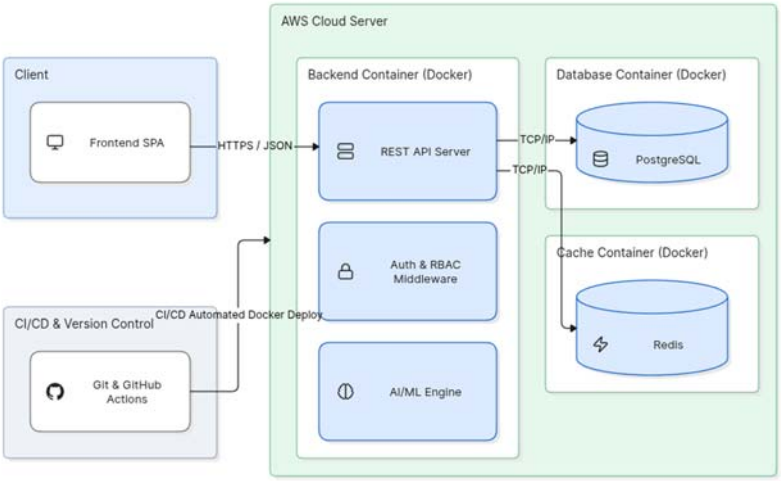


Рисунок 2. – Діаграма розгортання вебзастосунку

В майбутньому планується інтегрувати до систему велику мовну модель, що може доповнити процес обробки даних засобами ШІ та додатково консультувати користувача з питань роботи вебзастосунку.

Список використаних джерел

1. Vuković D. B., Dekpo-Adza S., Matović S. AI integration in financial services: a systematic review of trends and regulatory challenges. *Humanities and Social Sciences Communications.* 2025. Vol. 12. Article: 562. DOI: https://doi.org/10.1057/s41599-025-04850-8

2. Single Page Application. URL: https://www.geeksforgeeks.org/reactjs/what-are-single-page-apps/ (дата звернення: 19.10.2026).

3. Introduction to JSON Web Tokens. URL: https://www.jwt.io/introduction#what-is-json-web-token (дата звернення: 19.10.2026).

УДК 004

Віктор Раленко
старший викладач кафедри
інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євгеній Стоєв
старший викладач кафедри
інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

АРХІТЕКТУРНІ ПРИНЦИПИ СТВОРЕННЯ ВЛАСНИХ БІБЛІОТЕК РЕАКТИВНИХ КОМПОНЕНТІВ ВЕБІНТЕРФЕЙСІВ

У сучасній веброзробці швидкість створення та візуальна цілісність користувацьких інтерфейсів безпосередньо залежать від якості обраної системи компонентів. Попри наявність готових рішень на ринку, таких як Material UI або Ant Design, розробники дедалі частіше стикаються з необхідністю створення власних кастомних UI-бібліотек. Це зумовлено вимогами до унікального брендингу, специфічної інтерактивної поведінки елементів, а також потребою мінімізувати обсяг стороннього коду в клієнтському застосунку. Створення власної бібліотеки компонентів у екосистемі React вирішує низку інженерних завдань:

– уніфікація візуальної мови: забезпечення єдиного стилю кнопок, полів вводу, модальних вікон та навігаційних блоків у межах усіх проєктів команди чи організації;

– інкапсуляція складної логіки станів: приховування внутрішньої реалізації анімацій, обробки подій фокусу та валідації всередині ізольованого компонента, надаючи розробнику лише декларативний API через props;

– оптимізація продуктивності: відмова від громіздких універсальних фреймворків на користь легкої, спеціалізованої кодової бази з повною підтримкою механізму Tree Shaking.

Традиційне використання Webpack для пакування UI-бібліотек вимагає складної конфігурації транспайлерів та завантажувачів стилів. У сучасній інфраструктурі оптимальним рішенням є використання

інструменту збірки Vite у спеціалізованому режимі Library Mode.[1] Такий підхід дозволяє одночасно генерувати вихідні бандли у форматах ES Modules та CommonJS, автоматично виносячи базові залежності у секцію peerDependencies у файлі package.json. Це гарантує, що фінальний застосунок не отримає дубльованих екземплярів React під час інтеграції бібліотеки. Особливу увагу при проєктуванні реактивних UI-бібліотек слід приділяти архітектурі стилізації. Поєднання утилітарного фреймворку TailwindCSS із компонентно-орієнтованим підходом дозволяє досягти високої швидкості розробки базової сітки та типографіки. Водночас для реалізації складних динамічних станів таких як, наприклад, ефектів стрибка, пульсації або плавної зміни геометрії елемента залежно від дій користувача, доцільно комбінувати утилітарні класи з ізольованими CSS-модулями або підходом CSS-in-JS. Для забезпечення гнучкості та зручності повторного використання компоненти бібліотеки повинні проєктуватися за такими принципами:

1) Композиційність: замість створення монолітних компонентів із десятками параметрів доцільно розділяти складні елементи на атомарні підкомпоненти, що комбінуються через властивість children.

2) Підтримка передачі посилань: кожен інтерактивний елемент бібліотеки повинен надавати батьківському застосунку прямий доступ до базового DOM-вузла, який критично важливий для керування фокусом, інтеграції зі сторонніми бібліотеками форм та забезпечення доступності.[2]

3) Керовані та некеровані стани: компоненти вводу повинні підтримувати як роботу із зовнішнім менеджером стану, так і автономну роботу за допомогою внутрішніх хуків.

4) Модульність експорту: структура вхідних точок бібліотеки має дозволяти імпортувати лише ті компоненти, які реально використовуються у проєкті, не завантажуючи зайві стилі та скрипти в пам’ять браузера.

Описані архітектурні принципи було покладено в основу власної програмної розробки – бібліотеки інтерактивних React-компонентів «JumpableReactiveUILib» (зареєстровано авторське право на комп’ютерну програму, свідоцтво № 147559 від 01.06.2026).[3] Основною концепцією бібліотеки є підвищення чутливості користувацького інтерфейсу через використання фізичних метафор (ефектів пружини, пружного відскоку та інерційного руху) за допомогою чистих CSS-мікроанімацій без залучення важких сторонніх JavaScript-рушіїв. Архітектура JumpableReactiveUILib об’єднує 20 ізольованих реактивних компонентів (інтерактивні кнопки, картки з

ефектом глибини, пружні перемикачі-тумблери, індикатори прогресу, поля вводу та модальні контейнери). Кожен компонент розміщено в окремій директорії з власним файлом логіки та локальною таблицею стилів. Така модульна організація забезпечує:

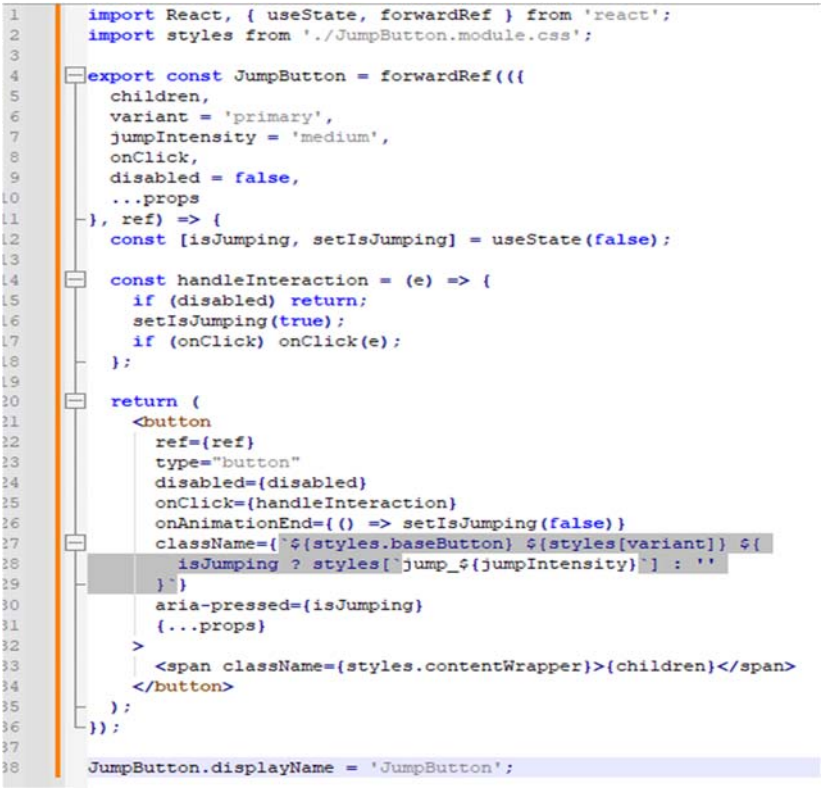
– високу продуктивність рендерингу: розрахунок траєкторій руху та трансформацій виконується безпосередньо рушієм браузера на рівні композиції шарів, що звільняє основний потік виконання JavaScript від зайвих перерахунків.

– семантичність та доступність: усі елементи побудовані на основі семантичних HTML5-тегів із повною підтримкою навігації з клавіатури та атрибутів aria-*, що робить інтерфейс інклюзивним.

– гнучкість інтеграції та масштабування: завдяки збірці через Vite у формат сумісного NPM-пакета, розробник може підключити як усю бібліотеку, так і окремий мікрокомпонент, передавши кастомні параметри амплітуди анімації, колірної теми та обробників подій через стандартний набір props.

На відміну від традиційних лінійних переходів, фізична правдоподібність компонентів у JumpableReactiveUILib досягається використанням кастомних кривих Безьє, де значення другої контрольної точки перевищує одиницю. Це створює ефект «перельоту» із подальшою стабілізацією елементу, імітуючи поведінку реальної механічної пружини без необхідності покрокового обчислення фізики у JavaScript.

Для запобігання деградації частоти кадрів на мобільних та слабких клієнтських пристроях усі анімації в бібліотеці суворо обмежені властивостями transform та opacity. Це дозволяє браузеру передавати рендеринг анімованих вузлів безпосередньо графічному процесору на етапі композиції шарів, повністю уникаючи дорогих операцій перерахунку геометрії макета та перемальовування пікселів. Крім того, на рівні CSS реалізовано підтримку медіа-запиту @media (prefers-reduced-motion: reduce), який автоматично вимикає інтенсивні стрибки для користувачів із вестибулярними порушеннями або увімкненим режимом енергозбереження ОС. Приклад програмної реалізації реактивного компонента кнопки з управлінням станом пружного стрибка та конфігурації експорту в середовищі JumpableReactiveUILib наведено на рисунку 1.



```
import React, { useState, forwardRef } from 'react';
import styles from './JumpButton.module.css';

export const JumpButton = forwardRef(({
  children,
  variant = 'primary',
  jumpIntensity = 'medium',
  onClick,
  disabled = false,
  ...props
}, ref) => {
  const [isJumping, setIsJumping] = useState(false);

  const handleInteraction = (e) => {
    if (disabled) return;
    setIsJumping(true);
    if (onClick) onClick(e);
  };

  return (
    <button
      ref={ref}
      type="button"
      disabled={disabled}
      onClick={handleInteraction}
      onAnimationEnd={() => setIsJumping(false)}
      className={`${styles.baseButton} ${styles[variant]} ${
        isJumping ? styles[`jump_${jumpIntensity}`] : ''
      }`}
      aria-pressed={isJumping}
      {...props}
    >
      <span className={styles.contentWrapper}>{children}</span>
    </button>
  );
});

JumpButton.displayName = 'JumpButton';
```

Рисунок 1. – Приклад програмної реалізації

Завдяки використанню обробника onAnimationEnd для скидання реактивного стану isJumping, компонент гарантує завершення циклу CSS-анімації навіть при швидких повторних натисканнях, зберігаючи плавність візуального відгуку.

Висновок. Розробка власної бібліотеки реактивних UI-компонентів на базі зв'язки React, Vite та сучасних інструментів стилізації дозволяє значно підвищити швидкість та якість створення клієнтських вебзастосунків. Практична апробація запропонованих принципів під час створення програмного комплексу «JumpableReactiveUILib» підтвердила, що поєднання ізольованих CSS-мікроанімацій із модульною компонентною архітектурою дозволяє досягти високої інтерактивності інтерфейсу без втрати продуктивності. Завдяки правильній конфігурації зовнішніх залежностей, ізоляції стилів та

продуманому API компонентів, команди розробників отримують масштабований інструмент, який поєднує високу продуктивність нативного рендерингу з гнучкістю сучасної компонентної архітектури.

Список використаних джерел

1. Vite Documentation: Building for Production (Library Mode). URL: https://vitejs.dev/guide/build.html#library-mode (дата звернення: 18.10.2026).
2. React Reference Overview: Manipulating the DOM with Refs. URL: https://react.dev/learn/manipulating-the-dom-with-refs (дата звернення: 18.10.2026).
3. Комп'ютерна програма «JumpableReactiveUILib» : свідоцтво про реєстрацію авторського права на твір № 147559 / В. С. Раленко, Є. Д. Стоєв, І. О. Кандиба, С. В. Бондаренко ; дата реєстрації 01.06.2026.

УДК 004

Віктор Раленко
старший викладач кафедри
інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Марина Фаленкова
старша викладачка кафедри
інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ЗАСТОСУВАННЯ МЕХАНІЗМІВ УВАГИ ДЛЯ ДЕТЕКЦІЇ САРКАЗМУ В ТЕКСТІ

Однією з найбільш нетривіальних проблем сучасного сентиментального аналізу є розпізнавання фігуральної мови, зокрема сарказму та іронії. В умовах стрімкого зростання обсягів користувацького контенту у соціальних мережах, відгуках та месенджерах, значна частка негативних оцінок висловлюється імпліцитно. Традиційні методи обробки природної мови, що спираються на словники тональності або частотні характеристики

лексем неефективні для вирішення цієї задачі, оскільки вони аналізують поверхневу полярність окремих слів без урахування глибинного семантичного конфлікту речення.

З лінгвістичної та математичної точок зору сарказм можна формалізувати як навмисну семантичну інверсію, за якої поверхнева позитивна оцінка вступає у гостру суперечність із негативним прагматичним або ситуативним контекстом висловлювання [1]. Для побудови адекватної математичної моделі детекції доцільно класифікувати саркастичні конструкції за характером формування дисонансу на три основні типи:

1) Лексико-контрастний сарказм – базується на прямому зіткненні полярних за тональністю маркерів у межах одного речення або суміжних клауз;

2) Пропозиційний сарказм – не містить явних негативних лексем, а негативний зміст виводиться через опис об'єктивно небажаного факту або числової аномалії (наприклад, тривалий час очікування або висока частота збоїв);

3) Паралінгвістично-маркований сарказм – інверсія полярності задається виключно через невербальні символи: специфічну пунктуацію, іронічні лапки, капслок або емодзі, що суперечать текстовому змісту.

Нехай вхідне текстове повідомлення представлене послідовністю токенів. У класичних адитивних або лінійних моделях загальна тональність тексту оцінюється як агрегована функція локальних тональностей окремих слів, проте у саркастичних конструкціях, таких як наприклад: «Чудова оптимізація додатку, тепер він вилітає всього тричі на хвилину» наявність сильного позитивного маркера «чудова» призводить до хибної класифікації, якщо модель не здатна зіставити його з віддаленим негативним фактом («вилітає тричі на хвилину»). Рекурентні нейронні мережі частково вирішують проблему послідовного контексту, однак через лінійну природу обчислень та згасання градієнта вони часто втрачають семантичний зв'язок між полярними частинами складних або розлогих речень [2].

Для подолання цієї проблеми необхідний перехід до глобального моделювання семантичних залежностей за допомогою механізму внутрішньої уваги, який лежить в основі архітектур класу Transformer. Під час обробки тексту кожен токен послідовності проєктується у три векторні простори: запитів, ключів та значень. Матриця ваг уваги обчислюється за формулою скалярного добутку з масштабуванням [3]:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (1)$$

де $\sqrt{d_k}$ – розмірність векторів ключів. Елемент матриці кількісно визначає ступінь взаємного семантичного впливу між і-м та j-м токенами незалежно від лінійної відстані між ними у тексті.

У контексті детекції сарказму ключову роль відіграє архітектура багатомакушкової уваги, яка виконує паралельне проєктування векторів у h незалежних підпросторів [3]:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W^O \quad (2)$$

де кожна голова формує власний спеціалізований підпростір ознак, а W – підсумкова вагова матриця проєкції.

Така декомпозиція дозволяє системі одночасно аналізувати текст на кількох рівнях лінгвістичної ієрархії:

1) Синтаксично-орієнтовані голови уваги встановлюють граматичні зв'язки між об'єктом оцінки та його характеристикою;

2) Семантично-контрастні голови уваги фіксують аномально високі вагові коефіцієнти між токенами з діаметрально протилежною базовою полярністю.

Таким чином, умову виникнення семантичного дисонансу у багатовимірному векторному просторі можна описати через функцію контрастності C(T):

$$C(T) = \max_{i,j}\left(\alpha_{ij} \cdot |s(t_i) - s(t_j)|\right) \quad (3)$$

Якщо значення C(T) перевищує заданий поріг чутливості, це свідчить про наявність у реченні сильно пов'язаних за змістом, але протилежних за емоційним забарвленням компонентів. З точки зору геометрії багатовимірного простору, механізм уваги виконує динамічну трансформацію топології векторного подання.[4] Якщо на початковому шарі ембеддінгів апріорний вектор позитивного слова та вектор негативного контексту знаходяться у протилежних півпросторах відносно розділяючої гіперплощини класифікатора то під час проходження через послідовність шарів трансформера відбувається ітеративне зміщення вектора маркера. На l-му шарі оновлений контекстуалізований вектор обчислюється з урахуванням зваженого впливу оточення:

$$v_i^{(l)} = \text{LayerNorm}\left(v_i^{(l-1)} + \sum_{j=1}^{n} \alpha_{ij}^{(l)} W_V^{(l)} v_j^{(l-1)}\right) \quad (4)$$

Оскільки для саркастичного висловлювання вага зв'язку із негативною частиною речення є домінуючою, результуючий вектор здійснює кутовий поворот у просторі, переходячи із позитивного кластера у негативний. У результаті конкатенації виходів усіх паралельних голів уваги формується глибоко контекстуалізований вектор речення, у якому початкова позитивна вага слова-маркера повністю інвертується під впливом домінуючого негативного контексту.

Додатковою перевагою використання механізмів уваги при аналізі іронічних текстів із соціальних мереж можна вказати здатність інтегрувати невербальні та паралінгвістичні маркери, такі як специфічну пунктуацію, взяття слів у лапки, емодзі. У традиційних конвеєрах попередньої обробки тексту пунктуація та спецсимволи зазвичай видаляються як інформаційний шум, що неминуче руйнує структуру сарказму. Натомість сучасні алгоритми субтокенізації зберігають ці елементи як окремі повноцінні токени. У математичній моделі уваги такі паралінгвістичні токени виступають у ролі нелінійних модифікаторів, яким матриця Self-Attention призначає високі вагові коефіцієнти відносно сусідніх оціночних прикметників. Наприклад, наявність токенів лапок навколо позитивного слова або додавання саркастичного емодзі в кінці позитивного твердження формує стійкий патерн активації специфічної голови уваги, яка автоматично змінює знак локальної полярності висловлювання на протилежний [5].

Водночас практичне застосування глибоких моделей на основі уваги для детекції сарказму стикається з двома фундаментальними проблемами: явищем розмиття уваги на наддовгих послідовностях та високим ризиком хибнопозитивних спрацьовувань на складних нейтральних або змішаних відгуках, де користувач об'єктивно перераховує як переваги, так і недоліки об'єкта без жодної іронії. Для розмежування щирої амбівалентної оцінки та сарказму стандартної функції втрат перехресної ентропії виявляється недостатньо. Теоретично обґрунтованим шляхом вирішення цієї проблеми є модифікація цільової функції навчання шляхом введення додаткового регуляризаційного компонента контрастної уваги [5]:

$$\mathcal{L}_{total} = \mathcal{L}_{CE}(y, \hat{y}) + \lambda \cdot \mathcal{L}_{contrast}\left(A, C(T)\right) \quad (5)$$

де y та $\hat{y}$ – істинна та прогнозована мітки класу тональності, A – тензор матриць уваги всіх голів моделі, а λ – гіперпараметр балансування.

Введення штрафу під час зворотного поширення помилки примушує окремі голови уваги фокусуватися виключно на асиметричних семантичних конфліктах, ігноруючи стандартні протиставні конструкції зі сполучниками «але» чи «проте», які притаманні звичайним об'єктивним текстам.

Висновки. Теоретичний аналіз демонструє, що класичні методи сентиментального аналізу, засновані на локальній оцінці тональності слів, є концептуально непридатними для розпізнавання сарказму та іронії. Застосування механізму багатомакушкової уваги забезпечує глобальне охоплення контексту та дозволяє математично формалізувати явище семантичного дисонансу через аналіз розподілу вагових коефіцієнтів між різнополярними елементами висловлювання та ітеративне зміщення векторів у просторі. Доведено критичну роль збереження паралінгвістичних маркерів під час токенізації як прямих індикаторів інверсії полярності. Перспективою подальших досліджень є оптимізація запропонованих контрастних функцій втрат та застосування еволюційних методів структурного синтезу для автоматизованого підбору конфігурації голів уваги в умовах зашумлених текстових даних.

Список використаних джерел

1. Dubey P., Dubey P., Bokoro P. N. Unpacking Sarcasm: A Contextual and Transformer-Based Approach for Improved Detection. *Computers.* 2025. Vol. 14, No. 3. Art. 95. DOI: https://doi.org/10.3390/computers14030095
2. Barik K. LSTM-DGWO-based sentiment analysis framework for analyzing online customer reviews. *SciSpace.* 2023. DOI: https://doi.org/10.1155/2023/6348831
3. Arunarani A. R. Bidirectional LSTM and self-attention mechanisms based multi-label sentiment analysis. *International Journal of System of Systems Engineering.* 2025. Vol. 15, No. 1. P. 67–78. DOI: https://doi.org/10.1504/IJSSE.2025.144565
4. Prahasto G. S., Setiawan E. B. Twitter social media-based sentiment analysis using Bi-LSTM method with genetic algorithms optimization. Khazanah Informatika: *Jurnal Ilmu Komputer dan Informatika*. 2025. Vol. 11, No. 1. P. 20–26. DOI: https://doi.org/10.23917/khif.v11i1.3959
5. Shaw L., Ansari M. W., Tahir E. Adversarial natural language processing: overview, challenges, and policy implications. *Data & Policy*. 2025. Vol. 7. P. e64. DOI: https://doi.org/10.1017/dap.2025.10030

УДК 004.42

Анна Садова
здобувачка першого (бакалаврського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євген Давиденко
кандидат технічних наук, доцент,
завідувач кафедри інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ЦИФРОВІЗАЦІЯ МЕНЕДЖМЕНТУ КЛІНІНГОВИХ КОМПАНІЙ НА ОСНОВІ ПРОГНОЗУВАННЯ ТА ЕВРИСТИЧНОГО ПЛАНУВАННЯ

Цифровізація клінінгових послуг охоплює не тільки онлайн-замовлення, а й управління роботою виконавців, календарями виїздів і контролем виконаних завдань.

За даними Grand View Research, обсяг світового ринку клінінгових послуг у 2025 р. оцінено в 442,1 млрд дол. США, у 2026 р. – у 470,8 млрд дол. США, а до 2033 р. прогнозується його зростання до 770,8 млрд дол. США (рис. 1). Такі оцінки підтверджують економічну вагу галузі, однак не обов'язково корелюють з результатами опитування населення про користування послугами.

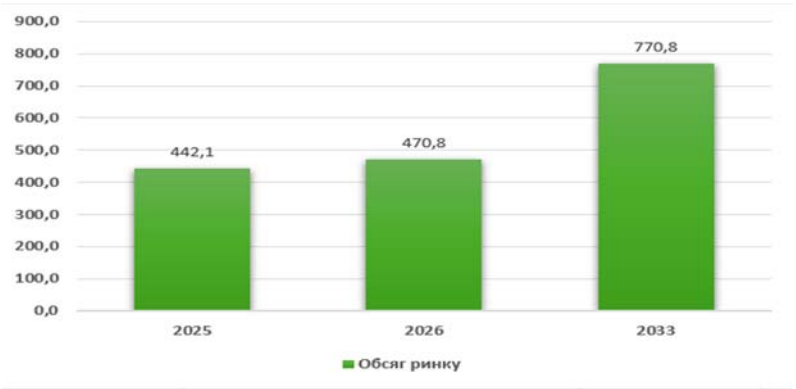


Рисунок 1. – Оцінки та прогноз обсягу світового ринку клінінгових послуг [1]

Проблема менеджменту клінінгової компанії полягає в узгодженні двох процесів: надходження замовлень від клієнтів і фактичної спроможності персоналу їх виконати. При ручному плануванні можливі накладання графіків, нерівномірна зайнятість працівників і затримки підтвердження заявок. Тому доцільно пов'язати прогнозування потоку замовлень із подальшим призначенням виконавців, замість розглядати ці задачі окремо.

Метою даної роботи є обґрунтування дворівневого підходу до цифровізації менеджменту замовлень і персоналу клінінгової компанії: попереднє визначення потреби в робочому часі та оперативне призначення працівників на конкретні замовлення. Підхід розглядається як функціональна складова веборієнтованої інформаційної системи, що об'єднує кабінети клієнта, виконавця й адміністратора.

На першому рівні пропонується застосовувати просте експоненційне згладжування (Simple Exponential Smoothing, SES) для короткострокового прогнозу кількості замовлень. Метод формує прогноз із більшою вагою нових спостережень [2]:

$$F_{t+1} = \alpha \cdot y_t + (1 - \alpha) \cdot F_t \qquad (1)$$

де: F_{t+1} – прогноз на наступний тиждень; y_t – фактична кількість замовлень у поточному тижні; F_t – прогноз поточного тижня; $\alpha \in (0; 1)$ – коефіцієнт згладжування.

Метод доцільний для коротких рядів без вираженої сезонності чи сталого тренду; за їх наявності необхідно розглядати складніші моделі [2].

Для ілюстрації припустимо, що фактично за тиждень надійшло 28 замовлень, попередній прогноз становив 22, а $\alpha = 0{,}4$. За формулою (1) прогноз наступного тижня дорівнює 24,4 замовлення. Якщо одне замовлення в середньому потребує 3 людино-години, очікуване навантаження становитиме близько 73,2 людино-години. За планового ресурсу 30 робочих годин на одного працівника потрібні щонайменше три працівники. Це лише попередня оцінка потужності: вона не враховує нерівномірність попиту за днями, переїзди між об'єктами та можливість одночасної роботи бригад.

На другому рівні використовується евристичний метод призначення замовлень за обмеженнями. Для кожної нової заявки система спочатку відбирає працівників, які мають потрібні навички,

доступні в зазначений часовий інтервал, не перевищують допустиме навантаження та можуть дістатися об’єкта вчасно. Серед допустимих кандидатів пріоритет отримує виконавець із найменшою кількістю вже запланованих робочих годин; за однакового навантаження – з меншим очікуваним часом доїзду. Подібна постановка задачі враховує обов’язкові обмеження й збалансований розподіл змін, характерні для планування персоналу [3].

Для перевірки послідовності прийняття рішення розглянуто модельне замовлення на 14:00. Дані щодо доступності, зайнятості та часу доїзду трьох працівників наведено у табл. 1. Умовні значення застосовано лише для демонстрації логіки відбору, а не як результати роботи реальної компанії.

Таблиця 1. – Модельний приклад вибору виконавця

Працівник	**Заплановано, год**	**Доїзд, хв**	**Доступний о 14:00**
А	22	15	Так
Б	16	25	Так
В	12	10	Ні

Згідно з табл. 1, працівник В виключається через часовий конфлікт. Із двох допустимих виконавців система рекомендує Б, оскільки його поточне навантаження менше, а прибуття на об’єкт залишається можливим. Адміністратор підтверджує або змінює рекомендацію; після підтвердження завдання відображається у кабінеті працівника.

Запропонований підхід не замінює рішення адміністратора, проте формує обґрунтовану рекомендацію. Інформаційна система повинна отримувати дату, тривалість і адресу замовлення, вид прибирання, статус заявки, графіки виконавців та історію виконаних робіт. На основі цих даних можливо формувати звіти про навантаження персоналу й, за окремо визначеними ставками, попередні внутрішні нарахування за виконані роботи.

Для подальшого оцінювання підходу доцільно визначити дві групи показників. Якість прогнозування слід перевіряти за середньою абсолютною похибкою на тижнях, не використаних для підбору коефіцієнта α. Якість планування – за кількістю конфліктів у розкладах, середнім часом призначення виконавця та різницею запланованих робочих годин між працівниками. Порівняння з ручним

розподілом дозволить з’ясувати практичну доцільність автоматизації без необґрунтованих заяв про вже досягнуте підвищення ефективності.

Таким чином, поєднання короткострокового прогнозування із призначенням виконавців за обмеженнями дозволяє перейти від простого обліку заявок до підтримки управлінських рішень. На цьому етапі наведено концепцію та модельні розрахунки; висновки про скорочення часу планування або підвищення ефективності потребують перевірки після реалізації на фактичних даних. Подальше дослідження доцільно спрямувати на оцінювання точності прогнозів, урахування сезонності та порівняння автоматизованого і ручного розподілу замовлень.

Список використаних джерел

1. Cleaning Services Market (2026–2033): Size, Share & Trends Analysis Report. URL: https://www.grandviewresearch.com/industry-analysis/cleaning-services-market-report (дата звернення: 18.10.2026).

2. Hyndman R. J., Athanasopoulos G. Forecasting: Principles and Practice. 3rd ed. Melbourne : OTexts, 2021. Chapter 8: Exponential smoothing. URL: https://otexts.com/fpp3/exponential-smoothing.html (дата звернення: 18.10.2026).

3. Employee Scheduling. Google OR-Tools Documentation. URL: https://developers.google.com/optimization/scheduling/employee_scheduling (дата звернення: 18.10.2026).

УДК 004.41

Дмитро Свінтицький
здобувач другого (магістерського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євген Давиденко
кандидат технічних наук, доцент,
завідувач кафедри інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ АВТОМАТИЗАЦІЇ ФОРМУВАННЯ ТА ОБЛІКУ НАВЧАЛЬНО-МЕТОДИЧНИХ МАТЕРІАЛІВ

Навчально-методичні матеріали (НММ) є основою підготовки персоналу навчально-тренувального центру. Їх створення поєднує роботу з навчальним змістом, реквізитами, структурою та різними видами документів. За файлового способу організації повторне використання однакових відомостей потребує ручного перенесення й зіставлення. Зміна автора, назви або структури матеріалу може вимагати виправлення кількох документів. Тому актуальною є автоматизація, яка пов’язує підготовку документа з обліком матеріалу та його складових.

Метою даної роботи є підвищення ефективності формування та обліку НММ шляхом розроблення програмного забезпечення зі структурованим зберіганням даних, формуванням документів за шаблонами й допоміжною перевіркою тексту. Для цього поєднано аналіз предметної області та програмних аналогів, моделювання процесів і даних, проєктування системи та визначення способів оцінювання її роботи.

Теоретичною основою є документна інженерія, що розглядає моделі документів у зв’язку з процесами використання інформації [1]. У запропонованому підході матеріал, документ і файл мають різне призначення. Матеріал є обліковою одиницею з назвою, типом, автором і навчальними цілями. Документ подає його у визначеній формі, а файл зберігає вихідний вміст або результат. Таке розмежування дозволяє використовувати спільні реквізити під час

підготовки кількох документів та знаходити їх через відповідний матеріал.

Робота автора починається зі створення картки матеріалу, визначення структури та приєднання вихідних фрагментів. Система перевіряє достатність даних для обраного виду документа, використовує шаблон і пов’язує результат із матеріалом. Наприклад, підготовка посібника потребує узгоджених заголовкових відомостей, вступу, змістовної частини, переліку розділів і літератури. Відсутню складову потрібно вказати авторові до завершення формування. Опис ролей, користувацьких сценаріїв і предметних зв’язків виконано із застосуванням елементів UML [2].

Архітектура передбачає вебінтерфейс і серверні сервіси з визначеною відповідальністю (рис. 1). Сервіс матеріалів веде облікові відомості й координує операції, сервіс формування документів працює з шаблонами та вмістом, довідникові сервіси надають пов’язані реквізити. База даних зберігає структуровані відомості та зв’язки, файлове сховище – вихідні й сформовані файли. Такий поділ дозволяє окремо розвивати способи підготовки документів і допоміжні функції, зберігаючи спільну предметну модель.

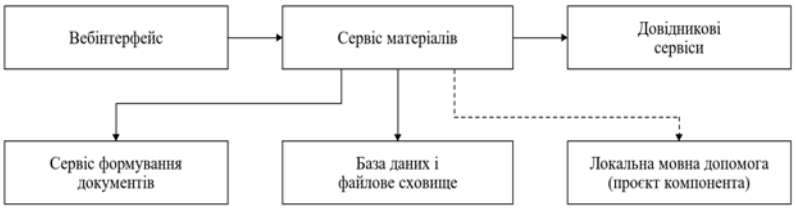


Рисунок 1. – Узагальнена організація системи автоматизації НММ

Вебінтерфейс організовується навколо завдань автора: пошуку матеріалу, редагування картки й структури, приєднання файлів та отримання документа. Користувач із правом перегляду працює з доступними результатами, адміністратор підтримує довідники й умови доступу. Після помилки система має пояснювати наступну дію та зберігати можливість виправлення вихідних даних. Основні складові рішення та їх призначення узагальнено в табл. 1.

Таблиця 1 – Основні складові програмного рішення

Складова	**Призначення**	**Результат для користувача**
Облік матеріалів	Картка, структура, пов’язані відомості	Пошук і повторне використання даних
Формування документів	Шаблон і узгоджені вихідні фрагменти	Документ, пов’язаний із матеріалом
Локальна мовна допомога	Рекомендації до вибраного тексту	Перегляд і прийняття змін автором

Допоміжна перевірка тексту доповнює основні функції системи. Підходи за правилами й нейромережеві методи пропонують різні способи виявлення та виправлення мовних помилок [3]. Для подальшого порівняння передбачено локальний коректор і компактну локальну модель. Автор отримує рекомендації до вибраного фрагмента та сам приймає зміни. Числові значення, одиниці вимірювання й спеціальні терміни потребують збереження технічного змісту. Розпізнавання сканів розглядається як можливість розвитку імпорту. Текст і зображення обробляються тільки локально.

Оцінювання системи передбачає перевірку наскрізних користувацьких сценаріїв, відповідності сформованих документів вихідним даним і коректного повідомлення про відсутні складові. Для оцінки користі автоматизації доцільно порівняти час підготовки та кількість ручних операцій за однакових умов. Мовні засоби потребують окремого зіставлення на українських навчальних текстах із перевіркою правильності пропозицій і збереження змісту. Ці показники визначають програму подальших випробувань.

В результаті запропоновано програмний підхід, що поєднує облік НММ, роботу зі структурою та формування документів зі спільних даних. Обґрунтовано предметну модель, організацію користувацьких сценаріїв і взаємодію сервісів; визначено місце локальної мовної допомоги. Практичне значення полягає у впорядкуванні підготовки матеріалів і створенні основи для зменшення повторних ручних дій. Подальша робота спрямована на завершення реалізації та підтвердження очікуваної ефективності під час випробувань.

Список використаних джерел

1. Glushko R. J., McGrath T. Document Engineering: Analyzing and Designing Documents for Business Informatics and Web Services. Cambridge, MA : MIT Press, 2005. 728 p.
2. Object Management Group. Unified Modeling Language. Version 2.5.1. URL: https://www.omg.org/spec/UML/2.5.1/About-UML (дата звернення: 20.10.2026).
3. Bryant C., Yuan Z., Qorib M. R., et al. Grammatical Error Correction: *A Survey of the State of the Art. Computational Linguistics.* 2023. Vol. 49, No. 3. P. 643–701. DOI: 10.1162/coli_a_00478

УДК 004.41:004.8

Євгеній Стоєв
старший викладач кафедри
інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Віктор Раленко
старший викладач кафедри
інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

АРХІТЕКТУРНІ КОМПРОМІСИ З AI: ОЦІНЮВАННЯ МАСШТАБОВАНОСТІ, НАДІЙНОСТІ, СКЛАДНОСТІ ТА ВАРТОСТІ

Проєктування корпоративного застосунку передбачає узгодження вимог, які можуть суперечити одна одній [1]. Незалежне масштабування компонентів потребує додаткової інфраструктури, резервування збільшує витрати, а декомпозиція системи ускладнює керування взаємодіями. Генеративний штучний інтелект (AI) можна залучати до формування альтернатив і систематизації їхніх наслідків [2]. Однак обґрунтованість вибору залежить від повноти контексту та перевірки припущень, на яких побудовані рекомендації моделі.

Метою даної роботи є обґрунтування підходу до оцінювання архітектурних компромісів із використанням AI за чотирма

взаємопов'язаними критеріями: масштабованістю, надійністю, складністю та вартістю.

Основою підходу є сценарний аналіз атрибутів якості за методом Architecture Tradeoff Analysis Method (ATAM) [1] та економічне зіставлення альтернатив за методом Cost Benefit Analysis Method (CBAM) [3]. Запропонований підхід має методичний характер; його ефективність потребує подальшого емпіричного оцінювання.

Початковим етапом є формування єдиного опису контексту: бізнес-операцій, меж доменів, інтеграцій, профілю навантаження, компетенцій команди та бюджетних обмежень. Якісні побажання переводяться у сценарії з вимірюваною реакцією системи [1]. Для операції оформлення замовлення визначають інтенсивність запитів, допустимий час відповіді та поведінку в разі недоступності зовнішньої служби. AI може допомогти виявити відсутні вимоги, проте цільові значення погоджують із зацікавленими сторонами.

Для порівняння AI отримує однакові вихідні умови та завдання сформувати кілька альтернатив: модульний моноліт, мікросервісну систему або рішення з вибірковим виділенням сервісів. Для кожного варіанта слід описати межі компонентів, володіння даними, синхронні й асинхронні взаємодії, спосіб розгортання та сценарії відмов [4]. Окремо фіксують невідомі параметри. Це дає змогу відокремити властивості запропонованої структури від неперевірених очікувань щодо її роботи.

Масштабованість оцінюють через зміну пропускної здатності, затримки та споживання ресурсів зі зростанням навантаження. Доцільно вимірювати p95 часу відповіді, тобто межу, якої не перевищують 95% запитів, і вартість оброблення одиниці навантаження. Незалежне масштабування сервісів може бути корисним за нерівномірного навантаження, але мережеві виклики та спільні залежності здатні обмежувати цей ефект [4]. AI формує гіпотези про вузькі місця; перевіряють їх навантажувальними тестами прототипів.

Надійність розглядають на рівні завершення критичних бізнес-операцій. Критеріями є цілі доступності, допустимий час відновлення (RTO) та допустима втрата даних, виражена часовим інтервалом (RPO) [5]. Аналіз охоплює відмови бази даних, брокера повідомлень і зовнішніх інтеграцій. Резервування та дублювання ресурсів можуть підвищувати доступність, водночас збільшуючи вартість і складність експлуатації [6]. Повторні спроби потребують обмежень та ідемпотентності операцій, інакше можливі перевантаження або дублювання бізнес-дій [7].

Складність доцільно оцінювати окремо для розробки й експлуатації. Враховують кількість одиниць розгортання, міжкомпонентних контрактів, сховищ даних, технологій та ручних процедур відновлення. Додатковими показниками можуть бути час внесення наскрізної зміни та трудомісткість діагностики збою. Менший обсяг коду окремого сервісу сам по собі не означає спрощення всієї системи [4]. AI допомагає скласти перелік залежностей і сценаріїв змін, а команда перевіряє його за моделлю архітектури та прототипом.

Вартість визначають для однакового горизонту планування та зіставних сценаріїв навантаження. Сукупна вартість володіння охоплює розробку, міграцію, обчислювальні ресурси, зберігання й передавання даних, ліцензії, моніторинг і супровід. Витрати на AI включають підготовку контексту, використання моделі та перевірку її результатів. Замість одного точного прогнозу доцільно розглядати діапазони для базового й пікового навантаження. Економічний аналіз має враховувати також очікувані наслідки простою, уникаючи подвійного обліку тих самих втрат [3].

Для узгодження критеріїв пропонується спочатку відсіяти альтернативи, які порушують обов'язкові обмеження безпеки, цілісності даних або бюджету. Далі формують матрицю показників із зазначенням джерела кожної оцінки: вимірювання, розрахунок чи експертне припущення. Серед допустимих варіантів визначають Парето-ефективні: поліпшення одного критерію в них потребує погіршення іншого [8]. Якщо застосовують зважений рейтинг, шкали нормалізують, ваги погоджують із бізнесом і перевіряють стійкість вибору до їх зміни. Непідтвердженим даним задають діапазон невизначеності.

Умовним прикладом є корпоративна система замовлень, у якій основні транзакції потребують узгодженого оновлення даних, а звітність створює нерівномірне навантаження. Порівнюються модульний моноліт, повна мікросервісна декомпозиція та виділення лише звітності в окремий сервіс. Останній варіант є кандидатом для перевірки, якщо бізнес допускає затримку оновлення звітів. Він потребує механізму надійного передавання змін і контролю актуальності даних. Вибір перевіряють за впливом звітності на транзакції, часом відновлення й витратами супроводу. Наведений приклад ілюструє процедуру аналізу та не є результатом експерименту.

Рекомендації AI потребують незалежної перевірки через ризик фактичних помилок і конфабуляцій, описаний у профілі генеративного

AI від NIST [9]. Для кожного суттєвого твердження слід вимагати перевірюване джерело або позначку про припущення. Повторне оцінювання тією самою моделлю може доповнювати аналіз, але не замінює інженерної валідації. До контексту передають лише дозволені проєктні дані. Прийняте рішення фіксують у записі Architecture Decision Record (ADR) з описом контексту, альтернатив, наслідків і статусу [10]; додатково зазначають докази та умови перегляду.

Додатковою умовою відтворюваності аналізу є збереження версії моделі, вхідного контексту, запиту та дати оцінювання. Якщо AI використовує проєктну документацію, кожне посилання має вести до чинної версії відповідного артефакту. Зміни вимог або профілю навантаження потребують повторної перевірки висновків. Це особливо важливо тоді, коли модель пропонує технологію без урахування наявної інфраструктури чи досвіду команди.

Валідація альтернатив має використовувати однакові набори даних і профілі навантаження. Окремо перевіряють штатну роботу, пікові запити та відновлення після відмови. Для асинхронної взаємодії вимірюють також час очікування в черзі й затримку актуалізації даних: швидке приймання запиту не дорівнює швидкому завершенню операції. Поряд із результатами фіксують конфігурацію середовища та витрачені ресурси, щоб різницю апаратних умов не приписати архітектурі.

Отже, запропонований підхід визначає місце AI у підготовці альтернатив і структурованому аналізі архітектурних компромісів. Вибір здійснюється серед рішень, які виконують обов'язкові вимоги, з урахуванням виміряних показників, економічних припущень і невизначеності. Практична перевірка підходу має охоплювати порівняння часу аналізу, кількості виявлених ризиків та відповідності прогнозів результатам випробувань із залученням AI і без нього.

Список використаних джерел

1. Kazman R., Klein M., Clements P. ATAM: Method for Architecture Evaluation. Pittsburgh : Software Engineering Institute, 2000. Report CMU/SEI-2000-TR-004. URL: https://sei.cmu.edu/library/atam-method-for-architecture-evaluation/ (дата звернення: 23.10.2026).

2. Dhar R., Vaidhyanathan K., Varma V. Can LLMs Generate Architectural Design Decisions? An Exploratory Empirical Study. 2024. arXiv:2403.01709. DOI: https://doi.org/10.48550/arXiv.2403.01709

3. Kazman R., Asundi J., Klein M. Making Architecture Design Decisions: An Economic Approach. Pittsburgh : Software Engineering Institute, 2002. Report CMU/SEI-2002-TR-035. DOI: https://doi.org/10.1184/R1/6575189.v1

4. Microservices architecture style. Microsoft Azure Architecture Center. URL: https://learn.microsoft.com/en-us/azure/architecture/guide/architecture-styles/microservices (дата звернення: 23.10.2026).

5. Architecture strategies for defining reliability targets. Microsoft Azure Well-Architected Framework. URL: https://learn.microsoft.com/en-us/azure/well-architected/reliability/metrics (дата звернення: 23.10.2026).

6. Reliability tradeoffs. Microsoft Azure Well-Architected Framework. URL: https://learn.microsoft.com/en-us/azure/well-architected/reliability/tradeoffs (дата звернення: 23.10.2026).

7. Timeouts, retries, and backoff with jitter. Amazon Web Services. 2019. URL: https://aws.amazon.com/builders-library/timeouts-retries-and-backoff-with-jitter/ (дата звернення: 23.10.2026).

8. Miettinen K. Nonlinear Multiobjective Optimization. New York : Springer, 1998. XXI, 298 p. DOI: https://doi.org/10.1007/978-1-4615-5563-6

9. Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile / C. Autio, et al. Gaithersburg : National Institute of Standards and Technology, 2024. NIST AI 600-1. DOI: https://doi.org/10.6028/NIST.AI.600-1

10.Nygard M. Documenting Architecture Decisions. Cognitect. 2011. URL: https://cognitect.com/blog/2011/11/15/documenting-architecture-decisions (дата звернення: 23.10.2026).

УДК 004.77:004.42

Анастасія Темна
здобувачка першого (бакалаврського) рівня вищої освіти
спеціальності «Інженерія програмного забезпечення»
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євгеній Стоєв
старший викладач кафедри
інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

РЕАЛІЗАЦІЯ КЛІЄНТСЬКОГО МОДУЛЯ ГЕОЛОКАЦІЙНОГО ПОШУКУ ТА РЕКОМЕНДАЦІЙ ОГОЛОШЕНЬ У СИСТЕМІ ДОПОМОГИ ВЕТЕРАНАМ “PORUCH”

В умовах повномасштабної війни та подолання її наслідків питання соціальної адаптації, реабілітації та побутової підтримки ветеранів війни і військовослужбовців з обмеженими можливостями набуває критичного значення. Часто представники цієї категорії населення стикаються з труднощами при виконанні базових повсякденних завдань – від придбання ліків та продуктів харчування до супроводу в медичні чи адміністративні установи. Такі потреби вимагають фізичного залучення сторонньої допомоги у безпосередній близькості до місця проживання заявника. Водночас волонтерська спільнота готова оперативно реагувати на подібні виклики, проте ефективність цієї взаємодії часто знижується через відсутність спеціалізованого цифрового інструменту, який би автоматично зв’язував запити з виконавцями за критерієм просторової близькості. У цьому контексті розробка інформаційної системи “Poruch”, що автоматизує процес пошуку та координації допомоги на основі геолокаційних даних, має високу практичну та соціальну цінність.

У сучасних наукових дослідженнях та практиці інженерії програмного забезпечення значна увага приділяється сервісам, орієнтованим на місцезнаходження користувача (Location-Based Services – LBS) [1]. Інтеграція алгоритмів просторової фільтрації безпосередньо у клієнтську частину вебзастосунку дозволяє суттєво підвищити релевантність видачі інформації, зменшити мережеве навантаження на сервер та скоротити час від моменту публікації

запиту до отримання реальної допомоги [2]. Використання сучасних реактивних фреймворків у поєднанні зі строгою статичною типізацією забезпечує високу швидкодію та відмовостійкість клієнтського інтерфейсу навіть при інтенсивній обробці масивів геоданих і динамічному оновленні станів компонентів [3; 4].

Метою даного дослідження є проєктування та програмна реалізація клієнтського модуля геолокаційного пошуку і формування рекомендованих оголошень у вебзастосунку ветеранської допомоги «Poruch». Загальна архітектура системи побудована на базі клієнт-серверної взаємодії. Серверна частина (Backend) реалізована з використанням платформи C# .NET і відповідає за бізнес-логіку, авторизацію, збереження анкет та обробку REST API запитів. Клієнтський застосунок (Frontend) розроблено на основі фреймворку Next.js та мови програмування TypeScript із залученням сучасних модульних UI-бібліотек для побудови доступного та інтуїтивно зрозумілого інтерфейсу (зокрема з урахуванням стандартів вебдоступності WCAG для людей з порушеннями зору чи моторики). У межах цих тез основну увагу зосереджено саме на алгоритмах роботи клієнтського модуля, який відповідає за визначення координат волонтера, розрахунок дистанцій та пріоритетну візуалізацію найближчих активних заявок.

Логіка роботи системи передбачає чіткий розподіл ролей та етапів обробки просторових даних. Коли ветеран або військовослужбовець з обмеженими можливостями створює заявку про допомогу, він вказує фізичну адресу виконання завдання у текстовому форматі (наприклад, місто, вулиця, номер будинку) або обирає точку на інтерактивній мапі. Під час ініціалізації оголошення введена адреса проходить процес прямого геокодування та перетворюється на пару географічних координат – широту (latitude) та довготу (longitude), які зберігаються у базі даних разом із деталями заявки.

Коли волонтер авторизується та відкриває головну сторінку застосунку на своєму пристрої, клієнтський модуль на Next.js за допомогою вбудованого браузерного інтерфейсу HTML5 Geolocation API асинхронно запитує дозвіл на отримання поточного місцезнаходження. Наприклад, якщо волонтер перебуває у Центральному районі міста, клієнтський скрипт фіксує його актуальні координати як базову точку відліку для формування персоналізованої стрічки оголошень.

Для забезпечення миттєвого відгуку інтерфейсу при перемиканні фільтрів та зменшення кількості надлишкових запитів до сервера на C# .NET, на клієнті реалізовано механізм попередньої просторової оцінки,

кешування та сортування отриманого пулу активних оголошень міста. Розрахунок найкоротшої відстані d між поточними координатами волонтера (ϕ_1, λ_1) та координатами адреси у заявці ветерана (ϕ_2, λ_2) здійснюється на клієнті за класичною формулою () гаверсинусів (Haversine formula), яка враховує сферичність Землі:

$$d = 2r\,arcsin\left(\sqrt{sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos(\phi_1)\cos(\phi_2)sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right) \quad (1)$$

де r – середній радіус Землі ($r \approx 6371$ км), а значення широти ϕ та довготи λ попередньо переводяться у радіани.

На основі обчисленого значення d клієнтський модуль здійснює категоризацію та ранжування заявок. Оголошення, які потрапляють у цільовий радіус від 1 до 1,5 кілометра від поточної позиції волонтера, автоматично отримують найвищий індекс релевантності. Детальну структуру розподілу заявок за радіусом дії та особливості їх відображення за допомогою UI-бібліотек наведено у таблиці 1.

Таблиця 1. – Класифікація та пріоритезація оголошень у клієнтському модулі залежно від радіуса пошуку

Категорія близькості	**Радіус від волонтера (км)**	**Пріоритет видачі у стрічці**	**Візуальне відображення в UI-інтерфейсі**	**Орієнтовний час прибуття волонтера**
Зона прямої доступності	до 1,0 км	Найвищий (Топ рекомендацій)	Спеціальний закріплений блок «Поруч з вами», акцентний колір рамки картки, бейдж «Піша доступність»	5-12 хвилин (пішки)
Зона рекомендованого охоплення	1,0-1,5 км	Високий (Блок рекомендацій)	Відображення у вкладці «Рекомендовані», індикатор точної дистанції у метрах	12-20 хвилин (пішки)
Районна зона	1,5-5,0 км	Середній (Загальна стрічка)	Стандартна картка оголошення у загальному списку після блоку рекомендацій	15-30 хвилин (громадський транспорт / авто)
Загальноміська зона	понад 5,0 км	Базовий (Доступно за фільтром)	Приховано з головної панелі рекомендацій, відображається при виборі фільтра «Усі заявки міста»	понад 30 хвилин

Використання строгої типізації TypeScript на цьому етапі дозволяє описати чіткі інтерфейси для моделей даних (зокрема IVeteranRequest, IGeoCoordinates, IRecommendationFilter), що унеможливлює виникнення помилок під час виконання коду при обробці асинхронних відповідей від сервера або при відсутності окремих необов'язкових полів у заявці. Крім того, застосування вбудованих механізмів мемоізації у React/Next.js (хуки useMemo та useCallback) гарантує, що перерахунок відстаней за формулою гаверсинусів для сотень оголошень відбувається лише тоді, коли координати волонтера змінилися на суттєву величину (понад 100 метрів), а не при кожному перемальовуванні компонентів сторінки.

Для забезпечення стабільності роботи під час руху волонтера містом у модулі реалізовано патерн відкладених викликів (debouncing). Якщо волонтер переміщується або вручну змінює радіус пошуку за допомогою UI-слайдера, оновлення списку рекомендованих оголошень відбувається з невеликою затримкою (300-500 мс) після завершення дії, що запобігає блокуванню основного потоку виконання браузера (Main Thread) та зайвим зверненням до API.

Водночас точність роботи клієнтського модуля суттєво залежить від апаратних можливостей пристрою користувача, наявності модулів супутникової навігації та політик конфіденційності сучасних веббраузерів. У реальних умовах експлуатації часто виникають ситуації, коли користувач випадково або свідомо блокує доступ до автоматичної геолокації, або ж сигнал GPS є нестабільним чи спотвореним через роботу засобів радіоелектронної боротьби (РЕБ) під час повітряних тривог. Для забезпечення безперервності роботи сервісу "Poruch" у клієнтському модулі спроєктовано трирівневу систему визначення координат (fallback-стратегію), порівняльну характеристику якої подано у таблиці 2.

Таблиця 2. – Порівняльний аналіз методів позиціонування та резервних сценаріїв (fallback) у клієнтському модулі

Метод отримання геоданих	**Джерело даних на клієнті**	**Середня похибка позиціонування**	**Вплив на формування рекомендацій (радіус 1-1,5 км)**	**Умови активації сценарію в системі "Poruch"**
Апаратний GPS / A-GPS	HTML5 Geolocation API (enableHighAccuracy: true)	5-30 метрів	Ідеальна точність, повна відповідність радіусу 1-1,5 км	Основний режим на смартфонах та планшетах при наданому дозволі
Мережева тріангуляція (Wi-	HTML5 Geolocation API	50-300 метрів	Висока точність, припустима	Використання застосунку з

Метод отримання геоданих	Джерело даних на клієнті	Середня похибка позиціонування	Вплив на формування рекомендацій (радіус 1-1,5 км)	Умови активації сценарію в системі “Poruch”
Fi / базові станції)	(стандартний режим)		похибка на межі радіуса 1,5 км	ноутбуків та ПК або слабкий сигнал GPS у приміщенні
Визначення за IP-адресою (GeoIP)	Сторонній сервіс / аналіз заголовків запиту на Next.js	1,5-10 кілометрів (рівень міста)	Низька точність (перевищує цільовий радіус), використовуєтьс я лише для вибору міста	Тимчасовий автоматичний режим при відмові у доступі до браузерної геолокації
Ручний вибір локації (Manual Fallback)	Інтерактивний UI-компонент вибору району або вулиці	0-200 метрів (залежить від вказаної точки)	Висока точність відносно обраного центру (наприклад, Центральний район)	Резервний режим при блокуванні геолокації або роботі засобів РЕБ

Як видно з наведених у таблиці 2 даних, у випадках, коли автоматична геолокація недоступна, покладатися на визначення місцезнаходження за IP-адресою неможливо, оскільки похибка перевищує цільовий радіус рекомендацій (1,5 км). Саме тому реалізований в інтерфейсі резервний сценарій ручного вибору (Manual Fallback) є критично важливим: волонтер може у два кліки обрати зі списку «Центральний район» або вказати перетин найближчих вулиць, після чого клієнтський модуль візьме координати цієї точки за еталон і миттєво сформує релевантний список заявок у радіусі 1-1,5 км від неї.

Перспективним напрямом подальшого розвитку розробленого модуля є інтеграція векторних картографічних бібліотек (зокрема Leaflet або Mapbox GL) для візуальної кластеризації маркерів заявок на мапі міста, а також врахування реальної довжини пішохідних маршрутів замість прямої сферичної відстані. Крім того, доцільним є впровадження технології двостороннього зв’язку WebSockets (за допомогою бібліотеки SignalR на боці сервера C# .NET) для миттєвого push-інформування волонтерів про появу нових термінових оголошень у їхньому радіусі перебування в режимі реального часу.

Таким чином, розроблений клієнтський модуль на базі фреймворку Next.js та мови TypeScript забезпечує ефективну, швидку та відмовостійку просторову фільтрацію заявок у інформаційній системі “Poruch”. Автоматичний математичний відбір оголошень у радіусі 1-1,5 кілометра від поточного місцезнаходження волонтера в поєднанні з продуманими резервними сценаріями позиціонування дозволяє

суттєво скоротити логістичні витрати часу, знизити бар'єр входу для нових добровольців та підвищити оперативність надання необхідної підтримки ветеранам війни та військовослужбовцям з обмеженими можливостями.

Список використаних джерел

1. Гавриленко О. В., Коваленко А. С. Розробка геоінформаційних модулів для сучасних вебзастосунків. *Вісник Національного технічного університету «ХПІ». Серія : Системний аналіз, управління та інформаційні технології.* 2023. № 1(9). С. 45–51.

2. Зайцев В. О. Оптимізація клієнт-серверної взаємодії у реактивних вебдодатках на базі Next.js. *Наукові праці Чорноморського національного університету імені Петра Могили. Серія : Комп'ютерні технології.* 2024. Том 350, Вип. 338. С. 72–78.

3. Freeman A. Pro TypeScript: Application-Scale JavaScript Development. 3rd ed. New York : Apress, 2023. 512 p.

4. Романенко Д. І., Литвин С. П. Архітектура кросплатформних систем волонтерської допомоги на базі .NET та сучасних JavaScript-фреймворків. *Проблеми програмування.* 2024. № 2. С. 64–73.

УДК 004.4

Микола Фісун
доктор технічних наук, професор,
професор кафедри інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Надія Гончарова
здобувачка третього (PhD) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ЗАСТОСУВАННЯ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ ДЛЯ ПЕРЕВІРКИ КОРЕКТНОСТІ ГРАМАТИК У ФОРМІ БЕКУСА – НАУРА

Вивчення курсу теорії формальних мов є критично важливим для фахівців з інформаційних технологій (ІТ), оскільки цей курс формує

глибоке розуміння фундаментальних принципів функціонування сучасних компіляторів та інтерпретаторів. Ці знання не лише дозволяють ефективно створювати власні мови програмування та інструменти статичного аналізу коду, а й значно підвищують абстрактне та системне мислення фахівця. Крім того, розуміння формальних мов допомагає краще опанувати регулярні вирази, методи обробки природної мови (Natural Language Processing, NLP) та алгоритми оптимізації, що робить фахівця IT більш гнучким при побудові складних архітектурних рішенні і розв'язанні складних алгоритмічних задач.

Важливою частиною процесу реалізації аналізаторів для формальних мов є створення форми Бекуса – Наура (Backus – Naur form, BNF). Форма Бекуса – Наура є метамовою, призначеною для опису синтаксису контекстно-вільних граматик, що робить її незамінним інструментом у проєктуванні мов програмування трансляторів. Ця метамова використовує спеціальні символи та правила заміни (продукції), щоб чітко й однозначно визначати структуру синтаксичних конструкцій – від змінних і операторів до складних блоків коду. Завдяки своїй строгості BNF дозволяє розробникам формально специфікувати граматику будь-якого формату даних чи мови, слугуючи надійною базою для створення автоматичних генераторів парсерів, що суттєво спрощує розробку та гарантує синтаксичну коректність програмного забезпечення.

Виконуючи індивідуальне завдання здобувачі освіти часто припускаються помилок, наприклад додають в граматику правила з лівою рекурсією для LL(1) аналізатору, або вписують неправильні породження в таблицю розбору. Ці помилки складно виявити, але їх вчасне виявлення та коректне виправлення необхідне для формування відповідних навичок у студентів.

У процесі вивчення формальних граматик необхідно використовувати вхідні мови, для яких створюватимуться аналізатори. Найчастіше як такі мови використовують предметно-орієнтовані мови (Domain-Specific Language, DSL). DSL призначені для використання в певній предметній області користувачами без поглиблених знань із програмування [1]. Через специфіку свого використання DSL має простіший синтаксис, ніж мови програмування загального призначення.

Прикладом предметно-орієнтованої мови є реляційна алгебра (РА). Реляційна алгебра є мовою запитів, розробленою Едгаром Коддом, та слугує теоретичною основою для реляційних баз даних і поширеної мови запитів SQL. Вона оперує відношеннями (таблицями) за

допомогою набору математичних операцій, таких як вибірка, проєкція, об'єднання, різниця, декартовий добуток та різні види з'єднань. Дослідження принципів роботи операторів реляційної алгебри є важливою частиною навчального курсу «Організація баз даних», що робить доцільним вибір саме цієї мови як вхідної мови для формальних граматик.

Граматику для реалізації оператора об'єднання (union) РА можна описати наступним представленням БНФ:

```
<union>::=<attribute_name>  UNION <attribute_name>;
<attribute_name>::=< braket> | <table_name>
<table_name>::=<symbol>{<symbol>}|<symbol>{<symbol> <number>}
<symbol>::=a|b|…|z
<number>::=0|1|2|…|9
<braket>::=(< braket_expr >)
<braket_expr>::=<minus>|<union>|<join>|<projection>|<extend>|
<semiminus>|<semijoin>|< divideby>|<braket>
```

У представленій граматиці відсутні неоднозначності, що дозволяє побудувати множини First та Follow і в результаті сформувати таблицю розбору. Ліва рекурсія – це ситуація в контекстно-вільних граматиках, коли породження нетермінального символу починається з цього самого нетермінального символу, що створює принципову проблему для LL(1) та інших спадних (top-down) аналізаторів. Оскільки LL(1)-парсер будує дерево синтаксичного аналізу зверху вниз і використовує лише один токен випередження для вибору альтернативи, натрапивши на ліву рекурсію, він потрапляє у нескінченний цикл рекурсивних викликів, так і не зіставивши жодного токена з вхідного потоку, що неминуче призводить до переповнення стека. Через це ліву рекурсію необхідно обов'язково усувати перед побудовою синтаксичного аналізатора шляхом еквівалентного перетворення граматики на граматику без лівої рекурсії [2].

Прикладом ліворекурсивних правил є:

```
<union>::=<union> UNION <attribute_name> | <attribute_name>;
<attribute_name>::=<braket> | <table_name>;
<table_name>::=<symbol>{<symbol>}|<symbol>{<symbol> <number>};
<symbol>::=a|b|…|z;
<number>::=0|1|2|…|9;
<braket>::=(<braket_expr>);
<braket_expr>::=<minus>|<union>|<join>|<projection>|<extend>|<semi
minus>|<semijoin>|<divideby>|<braket>;
```

Спростити процес перевірки робіт викладачем і виявлення типових помилок дозволяє використання великих мовних моделей. Великі

мовні моделі (Large Language Models, LLM) – це нейромережеві архітектури (найчастіше на основі трансформерів), навчені на великих обсягах текстових даних для інтерпретації, генерації та оброблення природної мови. До кола задач, які здатні розв'язувати LLM, належить аналіз граматик, представлених у вигляді БНФ.

Для реалізації програмного забезпечення з автоматизації перевірки практичних робіт з теми форми Бекуса – Наура необхідно підключити модель Gemini 3 Flash. Підключення API моделі Gemini 3 Flash у застосунках на Python здійснюється через офіційну бібліотеку google-genai, яка забезпечує швидкий і зручний доступ до високопродуктивних мультимодальних можливостей моделі. Процес починається з отримання персонального API-ключа в консолі Google AI Studio, після чого ключ налаштовується в середовищі як змінна оточення або передається безпосередньо в код [3]. За правильного налаштування мовна модель повідомить про наявні неоднозначності та надасть рекомендації щодо їх усунення (рис. 1).

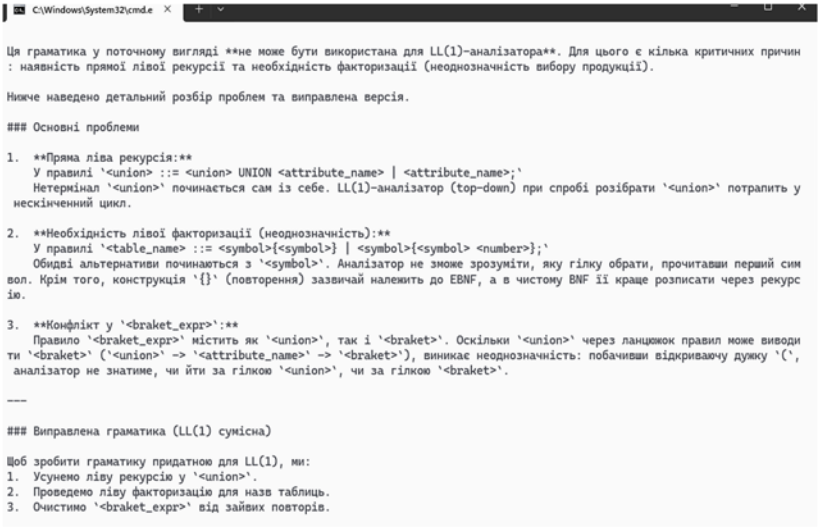


Рисунок 1. – Результат використання LLM Gemini 3 Flash

У майбутньому планується реалізувати перевірку лексичного та синтаксичного аналізаторів за допомогою LLM.

Список використаних джерел

1. Fowler M. Domain-Specific Languages. Pearson Education, Inc, 2010. 594 p.
2. Aho A., Lam M., Sethi R., et al. Compilers: Principles, Techniques, and Tools. Boston : Addison Wesley, 2006. 1040 p. ISBN 978-0321486813.
3. Кандиба І. О., Фісун М. Т. Аналіз журналів активності операційної системи Windows із використанням Python. *Таврійський науковий вісник. Серія : Технічні науки.* 2026. Том 1, № 2. С. 176–184. DOI: https://doi.org/10.32782/tnv-tech.2026.2.1.17

УДК 004.8+004.4’2

Аліна Худолій
здобувачка другого (магістерського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євген Давиденко
кандидат технічних наук, доцент,
завідувач кафедри інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

АНАЛІЗ ТОНАЛЬНОСТІ ВІДГУКІВ СПОЖИВАЧІВ МЕТОДАМИ МАШИННОГО НАВЧАННЯ

Емоційне забарвлення відгуків є одним із ключових джерел зворотного зв’язку для власників інтернет-магазинів і сервісних платформ. Якщо середня оцінка лише констатує загальну задоволеність, то тональність тексту розкриває її характер (подяку, розчарування або нейтральність). Понад 85% споживачів аналізують онлайн-відгуки перед придбанням товарів, тому точність їхньої автоматизованої обробки безпосередньо впливає на рішення покупця. Аналіз тональності зводиться до класифікації текстів за емоційною полярністю та є класичною задачею обробки природної мови.

Експериментальні дослідження виконувалися на наборі даних Jigsaw Toxic Comment Classification Challenge, що містить 159 571 коментар і 8 ознак у навчальній вибірці та 153 164 коментарі в

тестовій, що у сумі дає понад 300 тис. повідомлень. Джерелом даних є коментарі енциклопедії Wikipedia; пропущені значення відсутні, а структура даних придатна для задач обробки тексту.

Аналіз розподілу міток виявив різкий дисбаланс (рис. 1): найбільш поширена категорія toxic налічує 15 294 записи, тоді як найрідкісніша threat – лише 478, або 0,3% вибірки.

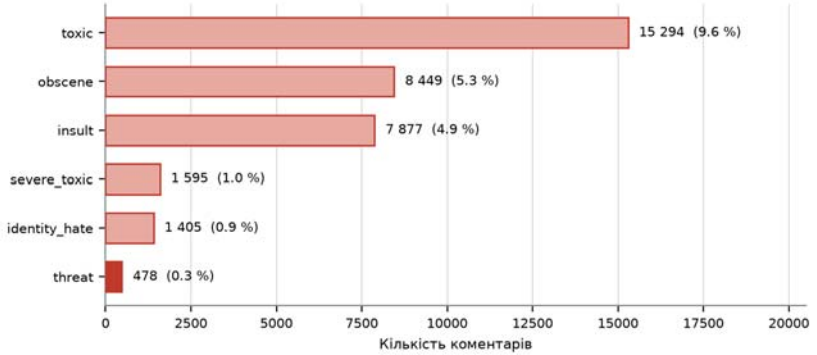


Рисунок 1. – Розподіл коментарів за категоріями
у навчальній вибірці

За такого дисбалансу модель схильна прогнозувати переважно домінантний клас, що знижує recall для рідкісних категорій і робить показник загальної точності непридатним для оцінки якості. Тому як основну метрику обрано F1-score у мікроусередненні, яке враховує внесок кожного класу пропорційно його частці, а зіставлення доповнено макроусередненням, що виявляє поведінку моделі на малих категоріях. Довжина коментарів також нерівномірна: середнє значення становить 394 символи, медіана не перевищує 205, а максимум досягає 5000 символів.

Текст було очищено від гіперпосилань, HTML-розміток і пунктуації та приведено до нижнього регістру, після чого застосовано TF-IDF з обмеженням у 10 000 ознак. Матриця розмірністю 159 571 × 10 000 є розрідженою, а найчастіше вживані слова корпусу (article, page, wikipedia) не несуть семантичного навантаження, що підтверджує доцільність зниження ваги службових токенів. Вибірка розподілена у співвідношенні 80/20, а додатково застосовано п’ятикратну крос-валідацію.

Для класифікації застосовано сім моделей різних типів – лінійні, ймовірнісні та стохастичні, обгорнуті у підхід One-vs-Rest, який дає

змогу розглядати багатокатегорійну задачу як набір незалежних бінарних задач. Результати наведено в табл. 1.

Таблиця 1. – Порівняльна оцінка моделей класифікації

Модель	**F1-score**	**Час навчання, с**
LinearSVM	0,718	5,34
SGD_hinge	0,704	2,34
LogReg	0,678	4,12
Perceptron	0,665	1,35
Ridge	0,645	7,73
SGD	0,586	1,68
NaiveBayes	0,581	0,40

Найкращий результат показав лінійний SVM із F1-score 0,718, за ним ішли стохастичний градієнтний спуск із hinge-функцією втрат (0,704) та логістична регресія (0,678). Перевірка лінійного SVM п'ятикратною крос-валідацією дала середнє значення F1-score 0,766 із розкидом менше 0,01, що свідчить про стабільність моделі. Агреговані метрики показали мікросереднє F1-score 0,72 проти макросереднього 0,56; для поширених класів F1-score сягає 0,77, а для рідкісних падає до 0,35-0,41 через низький recall. Перехід на біграми та налаштування регуляризації C = 2,0 додатково підвищили якість на 0,021, що підтверджує доцільність врахування контексту сусідніх слів.

Отримані експериментальні результати коректно переносяться на задачу аналізу тональності відгуків, оскільки структура розріджених векторів TF-IDF та закономірності поведінки класифікаторів є ідентичними.

Висновки: порівняльний аналіз семи моделей на 159 571 коментарі показав, що для класифікації розріджених ознак TF-IDF найефективніші лінійні моделі, а дисбаланс міток є головним чинником зниження якості на рідкісних категоріях: мікросередній F1-score 0,72 проти макросереднього 0,56. Застосування біграм і налаштування регуляризації дали приріст якості 0,021. Для задачі аналізу тональності відгуків це означає потребу балансування навчальної вибірки та обов'язкового супроводу результату пояснювальними ознаками. Подальші дослідження спрямовані на застосування федеративного навчання, за якого модель удосконалюється без передавання приватних даних.

Список використаних джерел

1. Pang B., Lee L. Opinion Mining and Sentiment Analysis. *Foundations and Trends in Information Retrieval.* 2008. Vol. 2, 1–2. URL: https://www.cs.cornell.edu/home/llee/omsa/omsa-published.pdf (дата звернення: 12.10.2026).
2. Sentiment Analysis. Stanford Natural Language Processing Group. URL: https://nlp.stanford.edu/sentiment/ (дата звернення: 12.10.2026).
3. Sentiment Labelled Sentences. UCI Machine Learning Repository. URL: https://archive.ics.uci.edu/dataset/331/sentiment+labelled+sentences (дата звернення: 12.10.2026).
4. Jigsaw Toxic Comment Classification Challenge. Kaggle. URL: https://www.kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge (дата звернення: 12.10.2026).
5. Feature extraction. scikit-learn documentation. URL: https://scikit-learn.org/stable/modules/feature_extraction.html (дата звернення: 12.10.2026).
6. Cross-validation: evaluating estimator performance. scikit-learn documentation. URL: https://scikit-learn.org/stable/modules/cross_validation.html (дата звернення: 12.10.2026).

УДК 004.02

Альона Швед
докторка технічних наук, професорка,
професорка кафедри інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Марина Фаленкова
старша викладачка кафедри
інженерії програмного забезпечення
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

СУЧАСНІ НАПРЯМКИ РОЗВИТКУ МЕТОДІВ АНАЛІЗУ ТОНАЛЬНОСТІ ТЕКСТІВ

Сучасні темпи цифровізації різних сфер суспільного життя людини неминуче ведуть до постійного збільшення обсягів інформації та ролі соціальних мереж, Інтернет-медіа, онлайн-відгуків та коментарів

користувача в сучасному суспільстві, зокрема інформаційному просторі. Щодня в мережі публікуються тисячі коментарів користувачів, постів, відгуків, формуючи цілу цифрову екосистему, що містить не лише об'єктивну інформацію, а й суб'єктивну оцінку автора, його емоції, думку та ставлення до різних подій. Наявність таких маніпулятивних маркерів чи емоційних тригерів дозволяє опосередковано чи безпосередньо впливати на сприйняття інформації та прийняття рішень, репутацію, медіа, настрій користувачів. У низці випадків користувацький контент, що характеризується певним емоційним забарвленням як засобом оперативного та стратегічного маскування, містить не лише маніпулятивну складову, а й дезінформаційні патерни.

У зв'язку з цим завдання аналізу тональності (сентимент-аналіз) стає все більш актуальним. Аналіз тональності (*Sentiment Analysis* / *Opinion Mining*) – це метод, що використовує технології обробки природної мови (*NLP*), машинного навчання та інтелектуального аналізу даних для виявлення, видобування та кількісної оцінки емоційного забарвлення або суб'єктивної складової (ставлення автора), що міститься в тексті [6, 8].

Метою даної роботи є систематичний огляд та критичний аналіз методів, сфер застосування та проблем, пов'язаних з аналізом тональності.

Аналіз емоційного забарвлення тексту та ставлення або настрою (намірів, психологічного стану) його автора, є цінним інструментом для бізнесу, державних установ та науковців, особливо в умовах реального часу. Результати аналізу тональності користуються попитом у різних сферах: у маркетингу та бізнесі вони використовуються для вивчення відгуків споживачів, оцінки репутації брендів, аналізу реакції цільової аудиторії на рекламну компанію або запуск нової лінійки товарів (послуг), відстеження громадської думки; у соціальних мережах та засобах масової інформації оцінка емоційного тону стрічок новин та коментарів, особливо в режимі реальної часу, допомагає дослідити громадську думку, відстежувати реакцію аудиторії, впливає на сприйняття конкретних персон та ситуацій, дозволяє прогнозувати та виявляти зміни в настроях користувачів; у сфері інформаційної безпеки методи сентимент-аналізу дозволяють виявляти маніпулятивний, провокаційний, фейковий контент; служба технічної підтримки використовує методи аналізу тональності для автоматичного оброблення вхідних звернень та оперативного реагування на емоційний стан клієнта.

В той же час ефективне виявлення ставлення користувачів серед великих масивів коментарів, постів та дописів є складним завданням. Актуальність аналізу тональності інформації визначається необхідністю автоматизованого вилучення та аналізу емоційно-оцінної складової з великих масивів текстових даних. Розробка та вдосконалення методів сентимент-аналізу являє собою важливе науково-практичне завдання, спрямоване на підвищення ефективності обробки неструктурованої текстової інформації (новин, відгуків, постів, та ін.), більш глибоке розуміння настроїв користувачів (виявлення прихованих емоційних забарвлень, відтінків відношення аудиторії), підтримку процесів прийняття рішень у різних галузях людської діяльності.

Наразі виділяють декілька напрямів аналізу тональності текстових описів [2, 4, 5-8]. Базовий (стандартний, полярний) аналіз тональності тексту – це підхід до оцінки емоційного забарвлення, що дозволяє класифікувати текст на декілька базових категорій (наприклад, різні рівні емоцій) [6]. Метод включає оцінку настроїв користувачів у межах заданої шкали. Найчастіше використовується двох (позитивний/негативний) або триполярні шкали (позитивна/негативна/нейтральна).

Градуйована або багатокласова тональність (*Fine-Grained Sentiment Analysis*) – це більш деталізований підхід до аналізу тексту, який розділяє тональність та визначає рівень емоційного забарвлення [2]. Як приклад можна навести 5-зіркову рейтингову систему оцінки попиту споживачів («дуже позитивно», «позитивно», «нейтрально», «негативно», «дуже негативно»), яка набула широкого поширення в маркетингу.

Аспектно-орієнтований аналіз (*Aspect-Based Sentiment Analysis*, *ABSA*) фокусується на аналізі тональності не лише тексту в цілому, а конкретних характеристичних властивостей аналізованого об'єкта (продукту, сервісу) чи події [4]. Наприклад, виробники телефонів опитують клієнтів щодо їх досвіду роботи зі звуком, графікою, сенсорною панеллю.

Аналіз емоцій (*Emotion Detection*) – це метод аналізу тональності, спрямований на розпізнавання емоційного стану і короткочасної реакції психіки людини під час написання тексту [5]. Підхід спрямований на виявлення та інтерпретацію різних емоцій, таких як радість, гнів, смуток, подив, через вибір слів та стилю написання текстового фрагмента.

Аналіз намірів (*Intent-Based Analysis*) спрямований на виявлення намірів (настроїв) цільової аудиторії [7]. Цей метод набув широкого

поширення у маркетингових дослідженнях для відстеження позиції певної групи клієнтів у циклі купівлі (наприклад, реакцію покупців на такі слова, як «знижки», «пропозиції» та «відгуки»; або виявляє намір здійснити покупку чи запит до служби підтримки).

Порівняльний аналіз (*Comparative Sentiment Analysis*) орієнтовано на вилучення та оцінку відношень між декількома об'єктами на основі текстових описів. На відміну від стандартного аналізу тональності (*Sentiment Analysis*), цей метод дозволяє оцінити перевагу одного об'єкта по відношенню до іншого (наприклад, «Продукт_А значно перевершує Продукт_Б за аналізованими характеристиками»).

Серед найбільш поширених підходів сентимент-аналізу можна виділити наступні [1, 3, 6, 8]. Підхід на основі правил (*Rule-Based Approach*) використовує заздалегідь визначені ключові слова на основі попередньо визначених словників (лексиконів) із позитивним і негативним забарвленням, граматичних конструкцій і правил для визначення тональності тексту. Головними недоліками цього підходу є недостатня гнучкість (масштабованість) у роботі з контекстом, неоднозначністю та мовними (культурними) особливостями.

Підхід на основі машинного (*Machine Learning*, *ML*) та глибокого (*Deep Learning*, *DL*) навчання передбачає використання алгоритмів та нейронних мереж, які навчаються на великих масивах розмічених текстів для розпізнавання складних закономірностей, контексту та сарказму. Основними недоліками такого підходу є залежність від якості та обсягу навчальних даних, значні витрати обчислювальних ресурсів, складність інтерпретації результатів і необхідність адаптації до різних предметних областей.

Гібридний підхід (*Hybrid Approach*) поєднує методи, що базуються на правилах, із глибиною аналізу, властивою машинному навчанню, що дозволяє з більшою точністю обробляти складні лексичні конструкції для отримання та оцінки контекстуального наміру у тексті. Попри існуючі переваги такого поєднання, він має низку недоліків, таких як складність архітектури, підвищені витрати на розроблення і обчислення, труднощі узгодження результатів різних методів та складність супроводу системи.

Висновки. В даній роботі представлено систематичний огляд завдань та методів аналізу тональності, досліджено сучасні напрямки його розвитку, проаналізовано сфери його застосування.

Аналіз емоційного забарвлення текстових даних дозволяє дати оцінку емоційного фону користувачів та їх ставлення до актуальних тем (події) та є джерелом цінної інформації для прийняття обґрунтованих рішень. Автоматизований аналіз тональності є

ефективним та швидким засобом оцінки емоційного забарвлення аудиторії, що дозволяє вчасно реагувати на зміни та коригувати обрану стратегію. Комплексне застосування методів обробки природної мови та машинного навчання дозволяє прискорити обробку інформації, підвищити ефективність сентимент-аналізу за рахунок більш точного виділення контексту, розпізнавання прихованої іронії, врахування культурного аспекту та комплексної оцінки емоційного забарвлення текстів. Значної ваги аналіз тональності набуває в умовах обробки великих масивів неструктурованих текстових даних, особливо в режимі реального часу (*Real-time Sentiment Analysis*), коли постає задача миттєвої обробки та оцінки емоційного забарвлення потокових текстових даних, де ручна обробка стає малоефективною та трудомісткою.

Список використаних джерел

1. Ashbaugh L., Zhang Y. (2024). A Comparative Study of Sentiment Analysis on Customer Reviews Using Machine Learning and Deep Learning. *Computers,* 13(12), 340. DOI: 10.3390/computers13120340
2. Chen Chao, Hu Xiaoli, Zhang Huibing, Zhaoyu Shou. (2020) Fine grained sentiment analysis based on Bert. Journal of Physics: *Conference Series,* 1651, 012016. DOI: 10.1088/1742-6596/1651/1/012016
3. Gupta Shailja, Ranjan Rajesh, Singh Surya. (2024). Comprehensive Study on Sentiment Analysis: From Rule-based to modern LLM based system. DOI: 10.48550/arXiv.2409.09989. URL: https://arxiv.org/pdf/2409.09989? (дата звернення: 17.10.2026).
4. Pallavi Shukla, Rakesh Kumar, Vijay Kumar Dwivedi, Ashutosh Kumar Singh. (2026) Aspect based sentiment analysis: A systematic review, taxonomy, applications, and future research directions. *Computer Science Review,* 61, 100924. DOI: 10.1016/j.cosrev.2026.100924
5. Pereira R., Mendes C., Ribeiro J., Ribeiro R., Miragaia R., Rodrigues N., Costa N., Pereira A. (2024). Systematic Review of Emotion Detection with Computer Vision and Deep Learning. *Sensors,* 24(11), 3484. DOI: 10.3390/s24113484
6. Saad Saidah, Saberi Bilal. (2017). Sentiment Analysis or Opinion Mining: A Review. *International Journal on Advanced Science, Engineering and Information Technology,* 7, 1660. DOI: 10.18517/ijaseit.7.5.2137
7. Xiaoang Zheng, Aris Leivadeas, Matthias Falkner. (2022) Intent Based Networking management with conflict detection and policy resolution in an enterprise network. *Computer Networks,* 219, 109457. DOI: 10.1016/j.comnet.2022.109457
8. Yanying Mao, Qun Liu, Yu Zhang. (2024). Sentiment analysis methods, applications, and challenges: A systematic literature review. *Journal of King Saud University – Computer and Information Sciences,* 36(4), 102048. DOI: 10.1016/j.jksuci.2024.102048

ПІДСЕКЦІЯ
ІНТЕЛЕКТУАЛЬНІ ІНФОРМАЦІЙНІ СИСТЕМИ

УДК 004.9

Надія Болюбаш
кандидат педагогічних наук, доцент,
доцент кафедри інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

СЕМАНТИЧНА СЕГМЕНТАЦІЯ ДЛЯ ІНТЕЛЕКТУАЛЬНОГО МОНІТОРИНГУ СІЛЬСЬКОГОСПОДАРСЬКИХ УГІДЬ ЗА ДАНИМИ БПЛА

Інтелектуальний моніторинг сільськогосподарських полів є перспективним напрямом застосування комп'ютерного зору для автоматизованого виявлення просторових аномалій посівів. Ефективне управління сільськогосподарськими угіддями передбачає своєчасне виявлення відхилень у стані посівів, оцінювання просторового розподілу проблемних ділянок і визначення територій, для яких необхідне додаткове обстеження. Традиційні методи моніторингу, засновані на наземних спостереженнях, потребують значних часових і трудових ресурсів, особливо за необхідності регулярного обстеження великих площ. Використання безпілотних літальних апаратів (БПЛА) дає змогу отримувати детальні аерофотознімки угідь, які можуть бути використані для автоматизованого аналізу стану посівів. Одним із перспективних напрямів такого аналізу є застосування нейромережевих моделей глибокого навчання для виявлення аномальних просторових патернів на зображеннях.

У сучасних дослідженнях для моніторингу використовують нейромережеві моделі різних архітектур, однак для розв'язання задачі моніторингу сільськогосподарських угідь доцільним є застосування підходів на основі семантичної сегментації. Детальні аерофотознімки дають змогу ідентифікувати ознаки пересихання, дефіциту поживних речовин, пропусків під час сівби, скупчення бур'янів, порушень водного режиму та інших неоднорідностей, що можуть свідчити про проблеми у стані угідь (рис. 1). Водночас автоматизоване розпізнавання таких патернів ускладняється різноманітністю їхніх

форм і розмірів, низькою контрастністю, нерегулярними межами, дифузним характером та значною часткою фону.

Класифікація зображень дає змогу визначати наявність певних ознак, проте сама по собі не забезпечує їхньої просторової локалізації. Методи детекції локалізують аномалії за допомогою обмежувальних рамок, що не дає змоги точно відтворювати межі ділянок неправильної форми. Семантична сегментація передбачає визначення класу кожного пікселя зображення, що дає змогу детально відтворювати просторове розташування та піксельні межі аномалій різних класів. Тому для автоматизованого аналізу аерофотознімків сільськогосподарських угідь, на яких необхідно виявляти аномальні ділянки складної форми, доцільно застосовувати нейромережеві архітектури семантичної сегментації.



Рисунок 1. – Проблемні ділянки сільськогосподарських угідь на аерофотознімках [1]

Дослідження семантичної сегментації доцільно проводити з використанням набору даних Agriculture-Vision – спеціалізованого набору аерофотознімків сільськогосподарських угідь із піксельною розміткою агрономічних аномалій та фону [1]. Agriculture-Vision містить RGB-зображення, ближній інфрачервоний канал (NIR) і піксельні анотації аномальних ділянок. Він призначений для дослідження методів автоматизованого розпізнавання просторових патернів на полях і дає змогу оцінювати моделі за якістю сегментації аномальних ділянок та відтворенням їхніх просторових меж. Із 2020 року в межах CVPR проводяться змагання Agriculture-Vision Challenge, спрямовані на розроблення та оцінювання алгоритмів автоматизованого розпізнавання агрономічних патернів за допомогою семантичної сегментації [2].

Для семантичної сегментації сільськогосподарських угідь на Agriculture-Vision досліджувалися різні архітектури моделей глибокого навчання, зокрема FCN, DeepLabV3, DeepLabV3+, FPN, UPerNet, PSPNet, HRNetV2, HRNetV2+OCR і SegFormer. У порівняльному

дослідженні A. Tavera, et al. на валідаційній вибірці Agriculture-Vision базові конфігурації більшості розглянутих моделей продемонстрували значення mIoU у діапазоні 0,2167–0,3007: зокрема, для UPerNet отримано 0,2167, DeepLabV3 – 0,2213, FPN – 0,2283, DeepLabV3+ – 0,2310, PSPNet – 0,2326, HRNetV2 – 0,2539, а HRNetV2+OCR – 0,3007 [3]. Водночас SegFormer з енкодером MiT-B5 забезпечила mIoU = 0,4496, а запропонований авторами підхід, що поєднує аугментацію, узгодження прогнозів і адаптивне вибирання навчальних зображень, підвищив цей показник до 0,4641.

Результати Agriculture-Vision Challenge також демонструють потенціал комплексного вдосконалення моделей. Зокрема, у Challenge 2024 ансамбль із балансуванням класів, спеціальною аугментацією та постобробкою досяг mIoU = 0,547 на тестовій вибірці, посівши друге місце [4]. У Challenge 2022 команда PAII-RS отримала mIoU = 0,582, застосувавши ансамбль трансформерних моделей та додаткові методи оброблення й аугментації [5]. Проведений аналіз показує, що якість сегментації залежить не лише від вибору архітектури, а й від конфігурації енкодера, організації навчання, способів подолання дисбалансу класів та оброблення прогнозів. Загалом наведені результати дозволяють розглядати SegFormer як перспективну трансформерну архітектуру для дослідження семантичної сегментації агрономічних аномалій.

Архітектура SegFormer поєднує ієрархічний енкодер Mix Transformer (MiT), що формує ознаки на різних просторових масштабах, із компактним MLP-декодером, який об’єднує ці ознаки для побудови сегментаційної карти (рис. 2) [6]. Багатомасштабне подання ознак є важливим для аналізу аерофотознімків сільськогосподарських угідь, оскільки аномалії відрізняються розмірами, формою та просторовою структурою. Поєднання ознак різних масштабів створює передумови для розпізнавання як невеликих, так і протяжних неоднорідних ділянок, а компактний декодер сприяє зменшенню обчислювальних витрат.

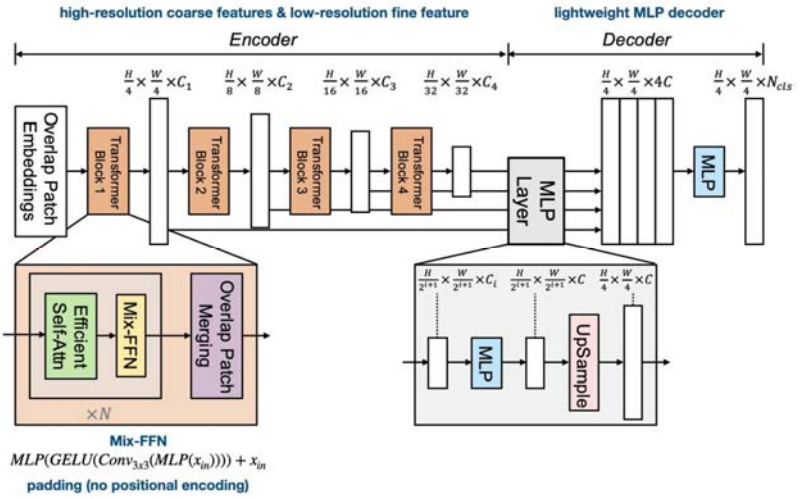


Рисунок 2. – Архітектура моделі SegFormer [5]

Механізм самоуваги (self-attention) в енкодері MiT дає змогу моделі SegFormer визначати значущість ознак різних просторових позицій для формування представлення поточної позиції. Завдяки цьому модель може враховувати не лише локальні текстурні характеристики, а й просторовий контекст. Наприклад, під час сегментації сільськогосподарських угідь зіставлення ознак різних ділянок може сприяти розпізнаванню водонасичення ґрунту, нерівномірності розвитку посівів або скупчень бур'янів.

Для обчислення самоуваги вхідні ознаки перетворюються на три матриці: Q (*Query*) – матрицю запитів, K (*Key*) – матрицю ключів і V (*Value*) – матрицю значень, які містять інформацію, що передається між просторовими позиціями. У SegFormer використовується багатоголовий механізм самоуваги (*Multi-Head Self-Attention, MHSA*), у якому кожна *j*-та голова обчислює увагу незалежно:

$$head_j = Soft\max\left(\frac{Q_j K_j^{\,T}}{\sqrt{d_k}}\right) V_j, \qquad (1)$$

де Q_j, K_j, V_j – матриці запитів, ключів і значень відповідної голови; d_k – розмірність вектору ключа.

Виходи всіх голів конкатенуються та перетворюються за допомогою вихідної лінійної проєкції.

Для зменшення обчислювальної складності SegFormer використовує ефективну самоувагу (*Efficient Self-Attention*), у якій просторові послідовності ключів і значень скорочуються, тоді як запити зберігають просторову роздільність відповідного рівня енкодера. У стандартних конфігураціях MiT-B0–B5 коефіцієнти скорочення для чотирьох рівнів становлять відповідно 64, 16, 4 та 1. Кількість голів на цих рівнях становить відповідно 1, 2, 5 та 8. Кількість Transformer-блоків на кожному рівні залежить від конфігурації енкодера.

Для оцінки якості семантичної сегментації в Agriculture-Vision застосовується модифікована метрика середнього перетину над об'єднанням (*modified mean Intersection over Union, mIoU*), яка враховує можливість наявності кількох правильних міток класів для одного пікселя. Її обчислюють за формулою:

$$mIoU = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{P_i + T_i + TP_i}, \qquad (2)$$

де N – кількість класів, включно з фоном; TP_i – кількість правильних передбачень класу i;

P_i – кількість пікселів, передбачених як клас i; T_i – кількість пікселів, анотованих як клас i.

Ці величини визначають за модифікованою матрицею помилок M, яка формується з урахуванням множинних правильних міток. Для кожного пікселя визначають x – передбачений клас, та Y – множину класів, наявних у його розмітці. Якщо передбачений клас належить множині правильних міток, тобто $x \in Y$, передбачення зараховують як правильне для кожного класу $x \in Y$:

$$M_{y,y} \leftarrow M_{y,y} + 1, \qquad \forall y \in Y. \qquad (3)$$

Якщо передбачений клас не належить множині правильних міток, тобто $x \notin Y$, для кожного правильного класу реєструють помилку:

$$M_{x,y} \leftarrow M_{x,y} + 1, \qquad \forall y \in Y. \qquad (4)$$

Дослідження семантичної сегментації було проведено з використанням моделі SegFormer-B1, вибір якої зумовлений доступними обчислювальними ресурсами. Навчання моделі здійснювалося на RGB-зображеннях сформованого на основі Agriculture-Vision 2021 збалансованого за класами набору даних, що містить вісім типів аномалій і фон – дев'ять класів. Клас endrow (кінець рядка) було виключено з цільових класів моделі, оскільки її розробка орієнтована на підтримку самостійного моніторингу угідь фермерами малих господарств, які мають безпосереднє уявлення про просторову конфігурацію власних полів. Клас storm_damage (пошкодження від шторму) також було виключено через його недостатню релевантність для визначених умов застосування моніторингу. За такої постановки задачі пріоритет надано автоматизованому виявленню аномалій стану посівів, що потребують додаткового обстеження.

На валідаційній вибірці було отримано значення стандартної метрики mIoU = 0,476. Водночас якість сегментації суттєво відрізнялася за окремими класами. Найвище значення IoU отримано для background (фон: 0,832), water (ділянки з водою: 0,634), далі – для double_plant (подвійний висів: 0,554), drydown (висихання посівів: 0,513). Нижчі значення IoU зафіксовано для класів weed_cluster (скупчення бур'янів: 0,231), waterway (водотоки: 0,306) і nutrient_deficiency (скупчення бур'янів: 0,304). Оцінювання моделі на незалежній тестовій вибірці показало значення стандартної метрики mIoU = 0,483 зі збереженням відмінностей у якості оцінювання сегментації різних класів. Вищі значення IoU отримано для класів background (фон: 0,861), water (ділянки з водою: 0,648), drydown (висихання посівів: 0,556), нижчі – для класів double_plant (подвійний висів: 0,369), weed_cluster (скупчення бур'янів: 0,286), waterway (водотоки: 0,352).

Результати проведеного дослідження демонструють перспективність застосування семантичної сегментації для інтелектуального моніторингу сільськогосподарських угідь за даними БПЛА шляхом піксельної локалізації аномальних ділянок. Набір Agriculture-Vision забезпечує основу для дослідження цієї задачі на спеціалізованих даних із відповідною розміткою, а архітектура SegFormer дає змогу формувати багаторівневі ознаки для сегментації об'єктів різного просторового масштабу. Проте отримані результати свідчать про неоднакову здатність моделі сегментувати різні типи аномалій і вказують на необхідність подальшого вдосконалення розпізнавання класів із нижчими показниками якості сегментації.

Перспективним напрямом подальших досліджень є використання більш потужних версій SegFormer з енкодерами MiT-B2–MiT-B5, застосування аугментації даних, використання модифікованої метрики mIoU, прийнятої в Agriculture-Vision Challenge, та аналіз можливостей практичного використання сформованих сегментаційних масок для моніторингу стану сільськогосподарських угідь.

Список використаних джерел

1. Chiu M. T., Xu X., Wei Y. et al. Agriculture-Vision: A Large Aerial Image Database for Agricultural Pattern Analysis. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020. P. 2828–2838. DOI: https://doi.org/10.1109/CVPR42600.2020.00290
2. Owais, M., Shafay, M., Zubair, M. et al. AgriVision: A Benchmark Dataset for Advancing Real-World Robotic Vision in Densely Fruited Blueberry Crop. *Scientific Data*. 2025. Vol. 12. 2014. DOI: https://doi.org/10.1038/s41597-025-06263-3
3. Tavera A., Arnaudo E., Masone C., Caputo, B. Augmentation Invariance and Adaptive Sampling in Semantic Segmentation of Agricultural Aerial Images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. 2022. P. 1656–1665. DOI: https://doi.org/10.1109/CVPRW56347.2022.00172
4. Liu W., Wang Z., Duan P., Kang X., Li S. Agriculture-Vision Challenge 2024 – The Runner-Up Solution for Agricultural Pattern Recognition via Class Balancing and Model Ensemble. 2024. DOI: https://doi.org/10.48550/arXiv.2406.12271
5. Yang Z., Lai J.-H., Zhou J., Zhou H., Du C., Lai Z. Agriculture-Vision Challenge 2022 – The Runner-Up Solution for Agricultural Pattern Recognition via Transformer-based Models. 2022. DOI: https://doi.org/10.48550/arXiv.2206.11920
6. Xie E., Wang W., Yu Z., Anandkumar A., Alvarez J. M., Luo P. SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers. *Advances in Neural Information Processing Systems*. 2021. Vol. 34. P. 12077–12090. URL: https://arxiv.org/abs/2105.15203 (дата звернення 27.10.2026).

УДК 519.6

Оксана Брагінець
кандидат фіз.-мат. наук, доцент,
доцент кафедри інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ЧИСЕЛЬНЕ РОЗВ'ЯЗАННЯ НЕЛІНІЙНИХ РІВНЯНЬ: ГРАФІЧНИЙ МЕТОД ВІДОКРЕМЛЕННЯ КОРЕНІВ ТА МЕТОД ХОРД У MAPLE

Абстракт. *Розглянуто особливості чисельного розв'язання нелінійних рівнянь із застосуванням графічного методу відокремлення коренів та методу хорд у середовищі Maple. Висвітлено теоретичні засади графічного дослідження функцій, що дозволяє визначити інтервали локалізації коренів та обирати початкові наближення для подальших обчислень. Розглянуто алгоритм методу хорд, його геометричну інтерпретацію та умови збіжності. Наведено програмні реалізації методів у середовищі Maple та продемонстровано їх використання для знаходження коренів нелінійних рівнянь.*

Вступ. Нелінійні рівняння відіграють ключову роль у сучасному математичному моделюванні, оскільки саме вони описують більшість реальних фізичних, технічних, біологічних та економічних процесів. На відміну від лінійних моделей, нелінійні залежності дозволяють відтворювати складні явища – від коливальних систем і хімічної кінетики до задач оптимізації та динаміки популяцій. Проте їхня аналітична розв'язність є винятком, а не правилом: у загальному випадку точний розв'язок або не існує, або має надто складний вигляд для практичного використання. Саме тому чисельні методи стають основним інструментом для знаходження коренів нелінійних рівнянь.

Основні труднощі розв'язування нелінійних рівнянь чисельними методами:

– відсутність універсального методу, який би гарантував розв'язок для всіх типів нелінійних рівнянь;

– залежність збіжності методів від вибору початкового наближення;

– необхідність контролю похибки та оцінки точності результатів;

– потреба у візуалізації процесу збіжності та аналізу поведінки функції.

Сучасні комп'ютерні математичні системи значно спрощують застосування чисельних алгоритмів, забезпечуючи високу точність, автоматизацію обчислень та можливість візуального аналізу. Одним із найпотужніших середовищ такого типу є Maple, який поєднує символьні та чисельні методи, підтримує широкий спектр алгоритмів для розв'язування нелінійних рівнянь і надає інструменти для аналізу збіжності, графічної інтерпретації та програмної реалізації власних методів. Такий підхід дозволяє показати практичну цінність Maple для ефективного навчання, досліджень та практичного застосування у різних галузях.

Загальні підходи до чисельного розв'язування нелінійних рівнянь. Розглядається рівняння вигляду:

$$f(x) = 0, \qquad (1)$$

де $f(x)$ – нелінійна неперервна функція однієї змінної, тобто $f: R^1 \wedge R^1$.

У загальному випадку рівняння (1) може мати багато коренів. Чисельні методи розв'язання нелінійних рівнянь, які розглянуто далі, дозволяють знаходити один корінь на заданому відрізку $[a, b]$. При цьому на інтервалі повинен існувати тільки один корінь. Знайти відрізок, що задовольняє цю умову можна різними способами:

а) з фізичних міркувань, тобто на основі фізичних знань про задачу;

б) на основі досвіду розв'язання аналогічних задач;

в) за допомогою графічних методів;

г) шляхом відокремлення коренів.

Якщо функція $f(x)$ заздалегідь відома, то найбільш ефективним є графічний спосіб пошуку відрізка $[a, b]$. В інших випадках, коли відрізок $[a, b]$ треба найти автоматично (не візуально), то застосовують алгоритм відокремлення коренів.

В чисельних методах рівняння вважається розв'язаним, якщо всі корені знайдені з заданою точністю, тобто з наперед заданою

похибкою. Знаходження наближених коренів рівняння звичайно складається з двох етапів:

1) відокремлення коренів, тобто знаходження для кожного з них відрізку ізоляції;

2) обчислення кореня у відрізку ізоляції з наперед заданою точністю.

Графічний метод. Графічний метод відокремлення кореня полягає у попередньому аналізі поведінки функції $f(x)$ на певному інтервалі з метою визначення ділянок, на яких функція змінює знак, а отже – гарантовано має корінь. Цей метод не дає точного значення кореня, але забезпечує початкову інформацію, необхідну для подальшого застосування чисельних методів (методу Ньютона, січних, простої ітерації тощо).

Ідея методу полягає в тому, що якщо функція неперервна на відрізку $[a, b]$ і виконується умова

$$f(a) \cdot f(b) < 0,$$

то, згідно з теоремою Больцано, на цьому відрізку існує принаймні один корінь рівняння (1). Графічний метод дозволяє візуально знайти такі відрізки, тобто побачити, де крива перетинає вісь Ox.

Переваги графічного способу:

– дозволяє гарантовано знайти відрізок, де є корінь;
– допомагає уникнути помилок при виборі початкових наближень;
– дає змогу виявити усі корені на заданому проміжку;
– забезпечує наочність і інтуїтивне розуміння поведінки функції;
– є універсальним і застосовується до будь-яких неперервних функцій.

Розглянемо реалізацію цього методу у Maple на прикладі рівняння

$$x^2 - 2 + e^{-x} = 0. \qquad (2)$$

Задаємо функцію та будуємо її графік (рис. 1):

```
> f := x → x^2 − 2 + e^(−x);
```

$$f := x \mapsto x^2 - 2 + e^{-x}$$

```
> with(plots) :
  plot(f(x), x = −3 ..3, color = black, thickness = 2, labels = ["x", "f(x)"]);
```

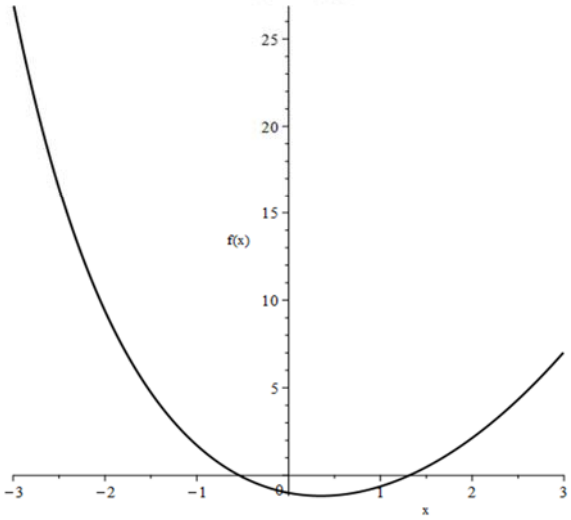


Рисунок 1. – Графічне відокремлення коренів рівняння (2) у Maple

Графік показує два перетини з віссю Ox на проміжках $[-1,0]$ та $[1,2]$, тобто рівняння має два корені. У Maple команда **fsolve** призначена для чисельного розв'язування рівнянь та систем рівнянь. На відміну від **solve**, яка шукає аналітичний розв'язок, **fsolve** повертає наближене числове значення кореня. Ця команда може знаходити один корінь за один виклик, використовуючи внутрішні алгоритми (модифікований Ньютона, комбіновані методи), тому важливо задавати інтервал, на якому треба відшукати розв'язок, якщо рівняння має не один корінь.

Оскільки рівняння (2) має два корені, тому використовуємо два виклики:

```
> x1 := fsolve(f(x), x = −1 ..0);
  x2 := fsolve(f(x), x = 1 ..2);
```

$$x1 := -0.5372744492$$

$$x2 := 1.315973778$$

Використання інтервалів, отриманих графічним методом, забезпечує стабільність та точність чисельного розв'язання. Далі продемонструємо застосування двох класичних методів для знаходження коренів на цих інтервалах.

Метод хорд. Метод хорд ще називають методом січних. Він базується на побудові хорди між двома точками $(a, f(a))$ та $(b, f(b))$ і знаходженні точки перетину цієї хорди з віссю Ox. Формула для нового наближення:

$$x_{k+1} = x_k - \frac{f(x_k)}{f(x_k) - f(c)}(x_k - c), \qquad k = 0, 1, 2, \ldots, \tag{3}$$

де $c \in [a; b]$.

Продемонструємо реалізацію методу хорд двома способами у Maple. Власна реалізація виглядає так:

```
> f := x -> x^2 - 2 + exp(-x);
  a := -1.0;
  b := 0.0;
  iter := 0 :
  epsilon := 10^-5;

  while (b - a) > epsilon do
    iter := iter + 1;
    c := b - f(b) * (b - a) / (f(b) - f(a));
      if f(c) * f(a) < 0 then
      b := c;
    else
      a := c;
    end if:
  end do:

  printf("Наближений корінь 1: x1 = %.7f (за %d ітерацій)\n", c, iter);
```

$$f := x \mapsto x^2 - 2 + e^{-x}$$

$$a := -1.0$$

$$b := 0.$$

$$\epsilon := \frac{1}{100000}$$

```
Наближений корінь 1: x1 = -0.5372744 (за 17 ітерацій)
```

```
> f := x → x^2 - 2 + exp(-x);
  a := 1.0;
  b := 2.0;
  iter := 0 :
  epsilon := 10^-5;

  while (b - a) > epsilon do
    iter := iter + 1;
    c := b - f(b) * (b - a) / (f(b) - f(a));
      if f(c) * f(a) < 0 then
      b := c;
    else
      a := c;
    end if:
  end do:

  printf("Наближений корінь 2: x2 = %.7f (за %d ітерацій)\n", c, iter);
```

$$f := x \mapsto x^2 - 2 + e^{-x}$$

$$a := 1.0$$

$$b := 2.0$$

$$\epsilon := \frac{1}{100000}$$

```
Наближений корінь 2: x2 = 1.3159738 (за 15 ітерацій)
```

Рисунок 2. – Програмна реалізація методу хорд у Maple

На кожному кроці перевіряється умова $f(c) \cdot f(a) < 0$, яка дозволяє звузити інтервал так, щоб він містив корінь. Завжди залишаємо той кінець інтервалу, де знак функції відрізняється від знаку в точці c. Це гарантує, що корінь залишиться всередині нового відрізка. Наприкінці виводить наближене значення кореня.

Використовуючи вбудовану команду **Secant** з пакета **Student[NumericalAnalysis]**, отримаємо:

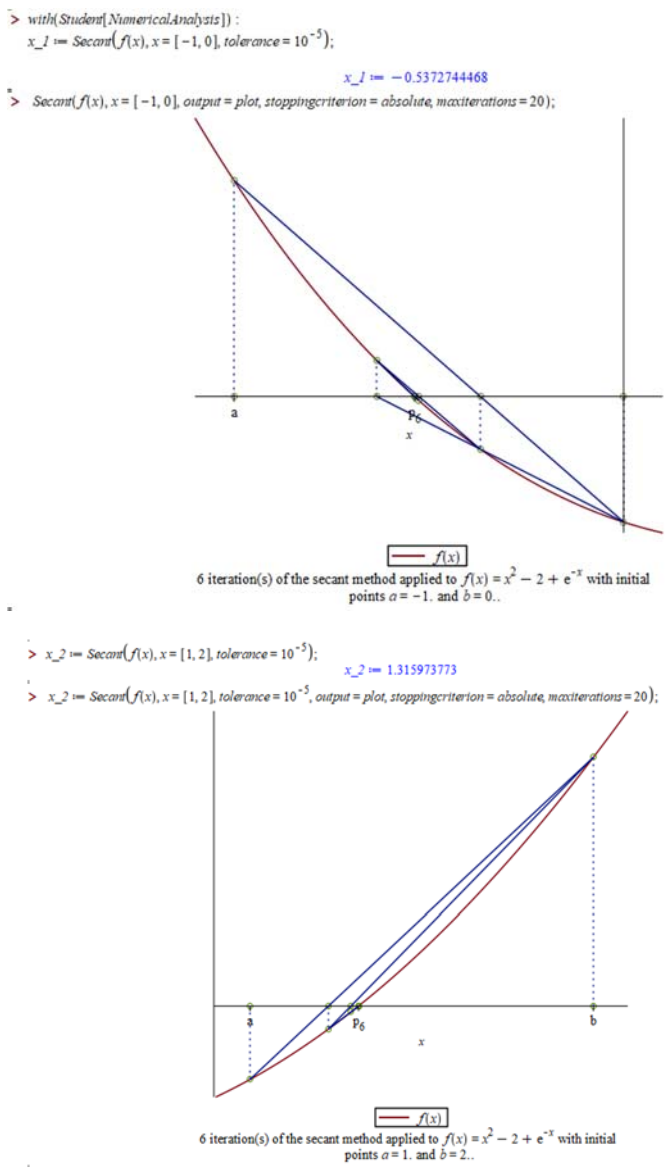


Рисунок 3. – Візуалізація методу хорд через вбудовану команду **Secant** у Maple

Власна реалізація дає змогу зрозуміти алгоритм методу покроково, а вбудована команда **Secant** зручна у простоті використання та швидкого отримання результатів у практичних задачах.

Висновки. У сучасних умовах розвитку комп'ютерної математики важливим є поєднання аналітичних та чисельних підходів до розв'язування математичних задач. Одним із таких підходів є використання графічних методів для локалізації коренів та чисельних методів для їх уточнення. Розв'язування нелінійних рівнянь є однією з ключових задач прикладної математики, і жоден метод не є універсальним. В роботі було проведено повний аналіз рівняння (2) із застосуванням графічного методу відокремлення кореня та уточнення коренів одним з чисельних методів, а також реалізовано всі алгоритми в Maple.

У [1] було реалізовано та досліджено чотири класичні методи: метод половинного поділу, метод хорд, метод Ньютона та метод простої ітерації.

Список використаних джерел

1. Брагінець О. (2026). Застосування Maple до розв'язування нелінійних рівнянь чисельними методами. *Комп'ютерно-інтегровані* технології*:* освіта*,* наука*,* виробництво*,* (63), 107–115. DOI: https://doi.org/10.36910/6775-2524-0560-2026-63-11

УДК 004.89:620.22-022.532

Олександр Герас
аспірант кафедри інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Олексій Козлов
доктор технічних наук, професор,
професор кафедри інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Едуард Лисенков
доктор фіз.-мат. наук, професор,
професор кафедри фізики та математики
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

НЕЙРОМЕРЕЖЕВІ МОДЕЛІ ДЛЯ ПРОГНОЗУВАННЯ ТЕХНОЛОГІЙ ПРИГОТУВАННЯ ТА ВЛАСТИВОСТЕЙ ПОЛІМЕРНИХ НАНОКОМПОЗИТІВ

Розглянуто перспективи застосування нейромережевих моделей для прогнозування технологій приготування та фізико-механічних властивостей полімерних нанокомпозитів. Проаналізовано повнозв'язні, багатозадачні, згорткові, графові та нейро-нечіткі моделі, а також попередньо навчені хімічні трансформери. За результатами порівняльного аналізу запропоновано послідовність вибору моделей залежно від доступних даних і схему їх поєднання з оптимізацією складу та оцінюванням невизначеності. Для прогнозування технологій приготування та властивостей нанонаповнених термопластичних композитів обґрунтовано доцільність порівняння багатозадачних повнозв'язних мереж, їх ансамблів та адаптивних нейро-нечітких систем.

Розроблення полімерних нанокомпозитів із заданими характеристиками потребує встановлення залежностей між складом, технологією приготування, мікроструктурою та властивостями. Підбір нанонаповнювачів, особливо їх поєднання, збільшує кількість можливих рецептур, а повний перебір складів і технологічних режимів

потребує значних експериментальних ресурсів. Нейромережеве прогнозування дає змогу попередньо оцінювати перспективність ймовірних складів композиту; практична цінність такого підходу визначається достовірністю прогнозу та його подальшою перевіркою на виготовлених зразках.

Для застосування полімерних нанокомпозитів значний інтерес становлять системи на основі термопластичних полімерів (поліаміду, полілактиду, полікарбонату) з вуглецевими нанотрубками, наноалмазами, арамідними нановолокнами та їх комбінаціями [1]. Вибір моделі для таких матеріалів залежить від типу вхідних даних, способу подання структури, кількості прогнозованих властивостей і можливості послідовного уточнення прогнозів. Поєднання цих критеріїв дає змогу визначити придатність нейромережевих підходів до обґрунтування складу та умов наступних експериментів.

Багатошаровий перцептрон (MLP) є початковим кандидатом для роботи з табличними даними про склад і параметри виготовлення. Глибокі повнозв'язні мережі застосовуються для прогнозування модуля пружності та міцності нанокомпозитів на основі поліетилену із шаруватим наповнювачем за його вмістом, вмістом допанта й параметрами приготування [2]. Перенесення такого підходу на інші полімери й наповнювачі потребує перевірки на відповідних експериментальних вибірках. Для обмеженої вибірки доцільно досліджувати компактні архітектури з регуляризацією.

Спільний блок мережі може формувати ознаки матеріалу, а окремі вихідні блоки – прогнозувати міцність, модуль пружності, ударну в'язкість і теплофізичні характеристики. Поєднання інформації про змішування та мікроструктуру дає змогу підвищувати точність прогнозування механічних властивостей, що підтверджено для композитів на основі гуми та технічного вуглецю [3]. Для досліджуваних термопластів такий підхід доцільно перевірити за наявності спільних закономірностей між цільовими показниками. За слабкого зв'язку між задачами спільне навчання може погіршувати окремі прогнози, тому потрібне порівняння з незалежними моделями.

Адаптивна нейро-нечітка система ANFIS поєднує нечіткі правила з навчанням їхніх параметрів на даних. ANFIS і штучні нейронні мережі застосовуються, зокрема, для прогнозування енергії руйнування полімерних нанокомпозитів [4]. У даному випадку ANFIS доцільно включити до порівняння як модель, що допускає подання залежностей у формі правил. Водночас зі збільшенням кількості входів зростає складність бази правил; обмежені дані потребують відбору ознак і контролю перенавчання.

Згорткові нейронні мережі (CNN) доцільні за наявності репрезентативних зображень мікроструктури. Вони дають змогу виділяти ознаки просторового розподілу частинок, які можна поєднувати з параметрами складу у спільній моделі. Поєднання скінченно-елементного аналізу з багатозадачними CNN дає змогу моделювати зв'язки між структурою і властивостями полімерних нанокомпозитів [5]. Важливо розрізняти навчання на розрахункових мікроструктурах і перевірку на реальних зразках: висока точність відтворення симуляцій сама по собі не підтверджує точності лабораторного прогнозу.

Графові нейронні мережі (GNN) застосовують до даних, у яких наявні істотні зв'язки між елементами структури. У моделі polyGNN молекулярний граф полімеру використано для формування ознак і багатозадачного прогнозування властивостей [6]. Молекулярний граф матриці та граф просторової організації наповнювача описують різні масштаби матеріалу. Тому застосування polyGNN до нанокомпозитів потребуватиме доповнення відомостями про наповнювач, його концентрацію, міжфазну взаємодію та технологію виготовлення. За фіксованої кількості матриць і відсутності структурних даних перевага складної графової моделі над табличною не є наперед визначеною.

Попередньо навчений хімічний трансформер polyBERT перетворює записи структури полімерів PSMILES на числові представлення, які використовуються для прогнозування властивостей [7]. Для нанокомпозитів перспективним є поєднання такого представлення матриці з характеристиками наповнювачів і режимів змішування. Це можливий спосіб перенесення знань про полімери, а не готова модель досліджуваних композицій. При роботі з трьома типами матриць його доцільність необхідно оцінювати відносно простішого кодування типу полімеру. Донавчання не усуває потреби в експериментальних даних щодо самих нанокомпозитів.

Для планування експериментів недостатньо лише точкового прогнозу властивості. Потрібно оцінювати, наскільки модель забезпечена даними в області прогнозованого складу. Одним із доступних підходів є ансамбль незалежно навчених нейромереж Deep Ensemble [8]. Розбіжність їх прогнозів характеризує компонент модельної невизначеності; її не слід автоматично ототожнювати з похибкою вимірювання або гарантованим довірчим інтервалом. Для практичного використання потрібні перевірка калібрування й оцінювання поведінки моделі на нових рецептурах.

Доцільно розділяти оптимізацію матеріалу та активне навчання. У першому випадку обирають склад із найкращим прогнозованим

поєднанням характеристик, у другому – експеримент, здатний найбільше уточнити модель. Байєсівська оптимізація поєднує прогноз із невизначеністю для послідовного вибору кандидатів; вона є методом планування пошуку, а не окремою архітектурою нейромережі. Для полімерних нанокомпозитів реалізовано одночасну оптимізацію складу й мікроструктури на основі гаусівських процесів і багатокритеріального пошуку [9]. Поширення такого підходу на прогнозування механічних властивостей обраних термопластів потребує окремої експериментальної перевірки.

Для початкових табличних даних пріоритетними кандидатами є компактна MLP, багатозадачна MLP та ANFIS. У свою чергу, ансамблювання додає можливість досліджувати невизначеність прогнозів. CNN доцільно залучати після формування набору мікроструктурних зображень, а GNN і polyBERT – за обґрунтованої потреби у врахуванні хімічної структури та перенесенні знань. Отже, перспективність моделі визначається відповідністю даним і завданню, а її складність має бути виправдана результатами незалежної перевірки.

Пропонується формувати початкову вибірку з контрольних зразків ненаповненого полімеру, однонаповнених композицій і композицій із поєднанням наповнювачів. За незмінного сумарного вмісту наповнювача варіювання співвідношення компонентів дасть змогу перевіряти ефект їх поєднання. Синергетичне поліпшення слід оцінювати відносно визначеного адитивного прогнозу та відповідних контрольних зразків з урахуванням розкиду вимірювань. Для кожної рецептури необхідно фіксувати марку матриці, характеристики наповнювачів, масові або об’ємні частки, поверхневу модифікацію, режими виготовлення й умови досліджень.

Після навчання та порівняння моделей пропонується застосовувати метаевристичні методи оптимізації для пошуку складів у технологічно допустимих межах. За суперечливих вимог до властивостей слід визначати множину Парето, тобто набір компромісних рішень. Рекомендовані до виготовлення склади мають охоплювати як прогнозовано перспективні рецептури, так і області, дослідження яких зменшують невизначеність. Результати випробувань використовуються для повторного навчання та коригування наступної серії експериментів. Критерієм ефективності такого циклу є досягнення заданих властивостей за визначеного експериментального бюджету.

Перевірку точності необхідно проводити з розділенням даних за рецептурами, партіями або джерелами публікацій. Повторні

вимірювання одного зразка та фрагменти одного мікрознімка не повинні потрапляти одночасно до навчальної й тестової частин. Нормування та налаштування моделей виконуються без використання незалежної тестової вибірки. Доцільно порівнювати нейромережі з градієнтним бустингом і гаусівським процесом за середньою абсолютною та середньоквадратичною похибками, стабільністю прогнозів і кількістю експериментів до досягнення цілі. Синтетичні дані слід позначати окремо; остаточна перевірка має спиратися на реальні зразки.

За результатами огляду запропоновано поетапний вибір нейромережевих моделей відповідно до доступних даних. Для прогнозування властивостей термопластичних нанокомпозитів обґрунтовано порівняння багатозадачних MLP, ансамблів і ANFIS; залучення CNN, GNN та polyBERT залежить від наявності структурної інформації. Перспективним напрямом є інтеграція прогнозування, оцінювання невизначеності та оптимізації складу в цикл послідовних експериментів. Запропонована схема становить методичну основу подальшого дослідження; скорочення експериментальних витрат і поліпшення властивостей мають бути підтверджені незалежною валідацією.

Список використаних джерел

1. Khan A., Iqbal S., Khan M., et al. A Comprehensive Review on Polymer Nanocomposites; Classification, Properties and Potential Applications. *Journal of Nanoscope.* 2023. Vol. 4, No. 1. P. 45–74. DOI: https://doi.org/10.52700/jn.v4i1.88
2. Zazoum B., Triki E., Bachri A. Modeling of Mechanical Properties of Clay-Reinforced Polymer Nanocomposites Using Deep Neural Network. *Materials.* 2020. Vol. 13, No. 19. Art. 4266. DOI: 10.3390/ma13194266
3. Wan Z., Chen S., Feng X., Sun Z.-Y. From processing to properties: Enhancing machine learning models with microstructural information in polymer nanocomposites. *Composites Communications.* 2024. Vol. 51. Art. 102072. DOI: 10.1016/j.coco.2024.102072
4. Hamdia K. M., Lahmer T., Nguyen-Thoi T., Rabczuk T. Predicting the fracture toughness of PNCs: A stochastic approach based on ANN and ANFIS. *Computational Materials Science.* 2015. Vol. 102. P. 304–313. DOI: 10.1016/j.commatsci.2015.02.045
5. Wang Y. et al. Mining structure–property relationships in polymer nanocomposites using data driven finite element analysis and multi-task convolutional neural networks. *Molecular Systems Design & Engineering.* 2020. Vol. 5. P. 962–975. DOI: 10.1039/D0ME00020E

6. Gurnani R., Kuenneth C., Toland A., Ramprasad R. Polymer Informatics at Scale with Multitask Graph Neural Networks. *Chemistry of Materials.* 2023. Vol. 35, No. 4. P. 1560–1567. DOI: 10.1021/acs.chemmater.2c02991

7. Kuenneth C., Ramprasad R. polyBERT: a chemical language model to enable fully machine-driven ultrafast polymer informatics. *Nature Communications.* 2023. Vol. 14. Art. 4099. DOI: 10.1038/s41467-023-39868-6

8. Lakshminarayanan B., Pritzel A., Blundell C. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. *Advances in Neural Information Processing Systems.* 2017. Vol. 30. URL: https://arxiv.org/abs/1612.01474

9. Iyer A. et al. Data centric nanocomposites design via mixed-variable Bayesian optimization. *Molecular Systems Design & Engineering.* 2020. Vol. 5. P. 1376–1390. DOI: 10.1039/D0ME00079E

УДК 004.8

Олександр Гожий
доктор технічних наук, професор,
професор кафедри
інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Ірина Калініна
доктор технічних наук, професор,
професор кафедри
інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ІНТЕЛЕКТУАЛЬНА СИСТЕМА ЕКОЛОГІЧНОГО МОНІТОРИНГУ З УРАХУВАННЯМ ДИСБАЛАНСУ КЛАСІВ

Актуальність. У сучасних умовах промислового розвитку та інтенсивного використання природних ресурсів особливого значення набувають питання екологічного моніторингу й забезпечення екологічної безпеки. Екологічний моніторинг – це система

спостереження, збору, обробки, передачі, зберігання та аналізу інформації про стан навколишнього середовища, а також прогнозування його змін і розробки науково обґрунтованих рекомендацій для прийняття рішень. Моніторинг стану довкілля та стійкості природних екосистем безпосередньо впливає на якість життя населення і здоров'я людей. До завдань екологічного моніторингу належать: контроль якості атмосферного повітря (викиди забруднювальних речовин), контроль якості води (стічні води, вплив на водні об'єкти), моніторинг стану ґрунтів (забруднення, ерозія), контроль шумового забруднення, енергоспоживання, а також моніторинг утворення та утилізації відходів.

Сучасний рівень розвитку інформаційних технологій дозволяє застосовувати інтелектуальні методи для аналізу великих обсягів екологічних даних. Такі підходи забезпечують автоматизацію процесів моніторингу, зокрема завдань класифікації. Однак під час роботи з реальними наборами екологічних даних виникає низка проблем, серед яких найважливішою є дисбаланс класів. Наявність дисбалансу класів у наборах даних є однією з основних перешкод при вирішенні завдань класифікації. Він виникає тоді, коли кількість прикладів одного класу суттєво перевищує кількість прикладів іншого. У реальних наборах даних такий перекіс спостерігається часто, оскільки аномальні події трапляються значно рідше, ніж звичайні. Дисбаланс в екологічних даних – наприклад, у даних про забруднення або аномальні екологічні ситуації – проявляється в різній кількості спостережень для нормальних і аномальних станів. Методи та алгоритми машинного навчання здебільшого орієнтовані на «мажоритарний» клас (клас, що переважає) і не здатні ефективно розпізнавати критичні випадки, які є важливими для завдань екологічного моніторингу.

У зв'язку з цим особливого значення набуває розробка інтелектуальних систем екологічного моніторингу, здатних виявляти дисбаланс класів і враховувати його під час класифікації екологічних показників. Такі системи мають забезпечувати точне розпізнавання рідкісних, але небезпечних станів довкілля, що є ключовою умовою для своєчасного реагування та запобігання екологічним катастрофам. Створення систем екологічного моніторингу потребує впровадження інтелектуальних підходів, які не лише підвищують точність класифікації, а й враховують специфічні характеристики даних, притаманні природним процесам, – зокрема їхню нерівномірність, мінливість та неповноту. Актуальність досліджень проблем дисбалансу класів зумовлена необхідністю створення ефективних

інструментів для автоматизованого аналізу показників довкілля з використанням сучасних методів аналізу даних і машинного навчання.

Метою цієї роботи є дослідження процесу врахування дисбалансу класів у задачах екологічного моніторингу, а також розробка та впровадження структури інтелектуальної системи класифікації, що враховує різний ступінь дисбалансу класів у наборах даних про стан довкілля.

Теоретична частина. Для розробки структури інтелектуальної системи класифікації на етапі аналізу та попередньої обробки даних було виділено три групи методів балансування даних: методи надмірної вибірки (*oversampling*), що збільшують кількість прикладів меншого класу; методи недостатньої вибірки (*undersampling*), що зменшують кількість прикладів більшого класу; а також комбіновані методи. Використання поєднання методів надмірної та недостатньої вибірки для формування збалансованого набору даних дозволяє досягти рівноваги між цими двома підходами. Такий комбінований підхід спрямований на нівелювання недоліків кожного з методів із одночасним використанням їхніх переваг.

Ще одна стратегія збалансування даних – або врахування їхньої незбалансованості – полягає у використанні методів, що застосовуються на етапах моделювання та класифікації. У контексті розв'язання задач машинного навчання це безпосередньо стосується етапів моделювання та прогнозування.

Структурований підхід до використання методів балансування та розв'язання задач класифікації є основою для розробки інтелектуальної системи класифікації [1, 2]. Система складається з двох підсистем: підсистеми аналізу та попередньої обробки даних і підсистеми моделювання та класифікації (рис. 1). Перша підсистема містить модулі для виконання базових процедур попередньої обробки даних, модуль балансування даних, а також модуль розподілу підготовлених даних на навчальну та тестову вибірки. До базових процедур підготовки даних належать: виявлення та обробка пропущених значень, виявлення та обробка викидів, а також нормалізація даних і діагностика дисбалансу класів у наборі даних. Модуль балансування використовується лише за наявності дисбалансу класів у даних.

Підсистема моделювання та класифікації включає модуль, що реалізує базові методи класифікації, адаптовані до умов дисбалансу даних; модуль ансамблевого моделювання; а також модуль оцінювання якості класифікаторів із використанням спеціальних метрик (точність (precision), повнота (recall), F1-міра, специфічність

(specificity), збалансована точність (balanced accuracy), G-середнє (G-mean), коефіцієнт кореляції Метьюза (MCC), AUC-ROC).

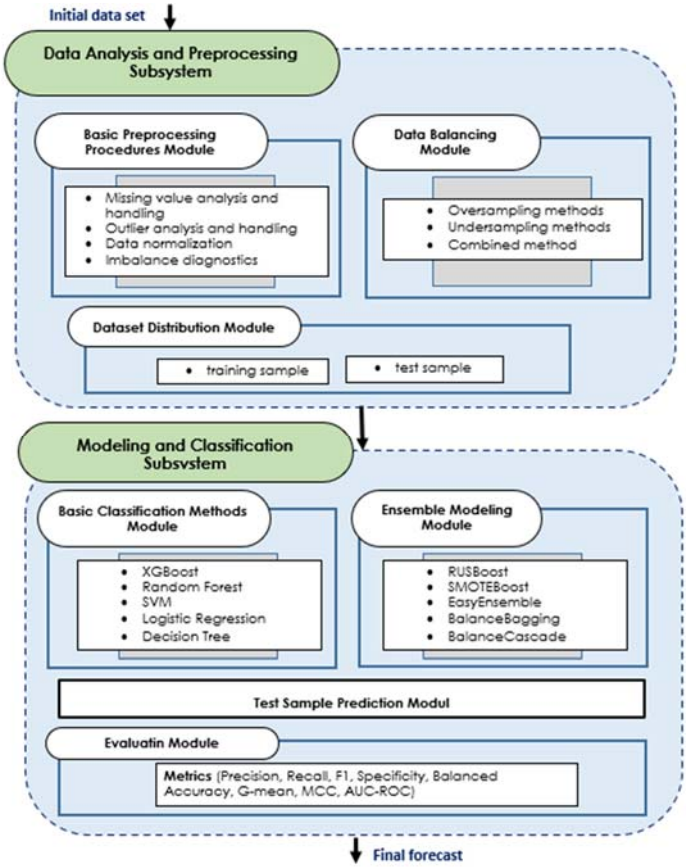


Рисунок 1. – Структура інтелектуальної системи класифікації

Ансамблеві методи ґрунтуються на поєднанні алгоритмів ансамблевого навчання з підходами, що базуються на модифікації даних, або з алгоритмами, чутливими до вартості помилок (cost-sensitive algorithms). У разі застосування підходу, орієнтованого на дані, у межах алгоритму ансамблевого навчання новий гібридний метод зазвичай передбачає попередню обробку даних перед навчанням кожного класифікатора. З іншого боку, ансамблі, чутливі до вартості

помилок, замість модифікації базового класифікатора для врахування «вартості» (ціни) помилок, реалізують процедуру мінімізації цієї вартості безпосередньо через алгоритм ансамблевого навчання. Вибір методів урахування дисбалансу класів у даних у межах обох підсистем залежить від ступеня та типу дисбалансу і відіграє вирішальну роль для якості кінцевої класифікації.

Висновки. Представлено системний підхід до розв'язання задач класифікації з урахуванням дисбалансу класів. Розглянуто проблеми, пов'язані з дисбалансом класів, та проаналізовано його різновиди. Розглянуто основні концепції й методи вирішення проблеми дисбалансу класів. Розроблено покроковий системний підхід до розв'язання задач класифікації. На основі цього підходу можливо створювати інтелектуальні системи класифікації, призначені для аналізу даних, їх попередньої обробки, балансування вибірок, а також побудови моделей машинного навчання для прогнозування.

Застосування системного підходу та методів балансування є ключовими чинниками підвищення якості класифікації в реальних проектах, де спостерігається дисбаланс класів.

Список використаних джерел

1. Kalinina I., Gozhyj A., Bidyuk P., Gozhyj V. Multilevel Ensemble Approach in Classification Problems. 2024 *IEEE 19th International Conference on Computer Science and Information Technologies (CSIT). Lviv, Ukraine, 2024.* DOI: https://doi.org/10.1109/CSIT65290.2024.10982625

2. Kalinina I., Gozhyj A., Gozhyi V., Shiyan S. Improving the architecture of a two-level heterogeneous ensemble for solving machine learning problems. ISW-2025: *Intelligent Systems Workshop at 9th International Conference on Computational Linguistics and Intelligent Systems (CoLInS-2025). Kharkiv, Ukraine, 2025.* URL: https://ceur-ws.org/Vol-3983/paper12.pdf (дата звернення: 28.10.2026).

УДК 004.8

Валерій Димо
викладач кафедри інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

БАГАТОЕТАПНЕ РОЗПІЗНАВАННЯ ПОШКОДЖЕНИХ БУДІВЕЛЬ НА СУПУТНИКОВИХ ЗНІМКАХ

Руйнування житлової, цивільної та іншої критичної інфраструктури України набувають все більшого масштабу, потенційно збільшуючись з кожним тижнем. Проведення оцінки руйнувань на знімках «вручну» займає неймовірну кількість часу, а безпосередній доступ до пошкоджених об'єктів неможливий через безпекову ситуацію або недоступність у зв'язку з окупацією. Оскільки оцінка пошкоджень як від військових дій, так й від масштабних руйнувань стихією, стикаються зі схожими проблемами, у світі вже давно використовуються напівавтоматичні системи на базі штучного інтелекту як Copernicus EMS [1].

Найпростіший підхід до задачі розпізнавання полягає у використанні однієї моделі для класифікації, виявлення об'єктів на знімку або його сегментації. Довгий час застосовуються такі архітектури як YOLO, Faster/Mask R-CNN [2], U-Net, Segformer, а з популярністю трансформерних архітектур поширилися потужні моделі, навчені на великій кількості даних, як DETR, DINO та SAM, які часто можуть виконувати декілька завдань одночасно (наприклад, локалізацію на сегментацію об'єктів). Водночас застосування лише однієї моделі для вирішення завдання розпізнавання пошкоджених будівель може не приносити необхідних результатів через комплексність проблеми: велике різноманіття будинків й типів ушкоджень, щільна забудова місцевості, а також складність якісної оцінки великих обсягів даних.

В рамах даного дослідження розглядається інтелектуальна технологія (рис. 1) з декількома етапами: автоматизованим формуванням набору даних, навчанням й донавчанням двох моделей (для локалізації та сегментації будівель відповідно), а також проміжними етапами, які потенційно нададуть можливість оптимізувати результати в залежності від визначених параметрів.

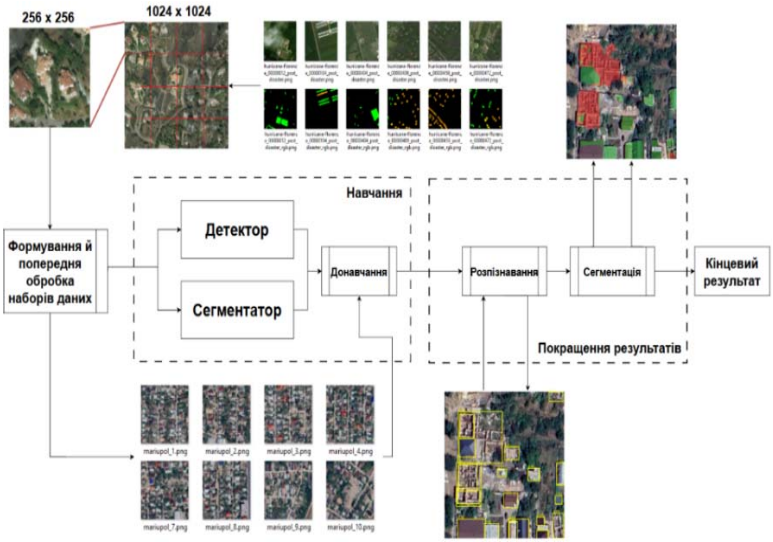


Рисунок 1. – Запропонована архітектура інтелектуальної технології

У поточному дослідженні було використано модель Faster R-CNN для розпізнавання (локалізації) будівель, та U-Net для їх сегментації; у подальшому можливе застосування й інших моделей, або навіть їх ансамблів, що потенційно може покращити результати або надати більше інформації для їх інтерпретування. Як можна побачити, на перших кроках моделі навчаються на окремому більшому наборі даних для успішного отримання попередніх узагальнених даних про будівлі, після чого відбувається донавчання на «спеціалізованому» наборі (будуть наводитися пізніше), що містить зображення забудови України, що покращить результати локалізації й сегментації у нашому домені.

Основний етап розпізнавання складається із локалізації областей, які були визначені детектором, що вирізаються із деяким відступом, після чого передаються сегментатору. Після сегментації усіх визначених областей результат вставляється у знімок назад, формуючи цілісну картину. Оскільки сегментація обмежується областю будівлі та її оточенням, модель мінімізує шанси хибного розпізнавання окремих структур як доріг або приватних ділянок (городів, дрібних архітектурних об’єктів тощо), оскільки відсоток класифікації цих об’єктів як будівель мінімальний.

Багатоетапність запропонованого підходу, окрім у застосуванні двох окремих моделей, полягає у можливому проміжному покращенні результатів шляхом оптимізації визначених параметрів, наприклад, порогових значень локалізації об'єктів, задання обмежень для рамок, визначення довірчих інтервалів відношення пікселів до заданих класів у випадку сегментації тощо.

Навчання проводилося з використанням двох наборів: автоматично сформованої підмножини набору даних xBD [3], що є одним із найбільших наборів даних для оцінки пошкоджених будівель, який наявний у відкритому доступі, а також власного набору даних, сформованого з використанням фотографій із сервісу Google Earth. Частина набору xBD містила 100 зображень 1024x1024 та 5 класів (бекграунд та 4 класи будівель), яка в свою чергу розбивалася на менші зображення 256x256, що пов'язано із технічними обмеженнями. Власний набір даних складався зі 100 зображень 256x256 і 4 класів (бекграунд та 3 класи будівель), які містили щільну забудову приватного сектору міста Маріуполь; взято нові зображення за 2024 рік, які є більш якісними відносно перших, зроблених у 2022 році. Для можливості застосування наборів із різною кількістю класів, проміжні класи пошкоджених будівель xBD були об'єднані в один.

На рис. 2 наведено деякі з результатів навчання й тестування моделі сегментації на обох наборах даних.

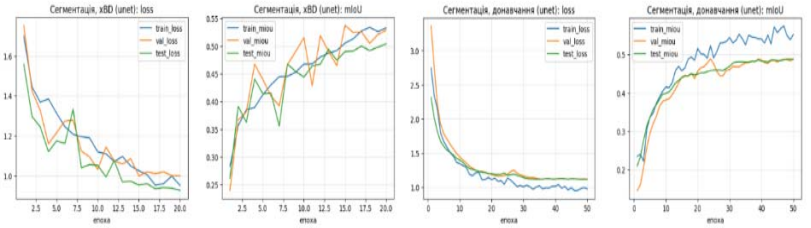


Рисунок 2. – Результати навчання моделі сегментації

Згідно результатів, базова версія U-Net, натренована на підмножині xBD, має ідентичні показники mIOU до версії U-Net із механізмом уваги і вагами ImageNet, яка була натренована лише на власному наборі даних із більшою кількістю епох. Двоетапне розпізнавання (рис. 3) показало гірші результати, враховуючи складність формування обмежувальних рамок при щільній забудові, і знизилося з приблизно 48% до 43%. В свою чергу, варто відмінити можливість оптимізації

визначення рамок, а також зниження «шуму» поза рамками, на відміну від простої сегментації. Частково негативні результати пояснюються й необхідністю збільшення рідкісних класів як у підмножині xBD, так і власному наборі даних.



Рисунок 3. – Приклад локалізації й сегментації пошкоджених будівель

Результати дослідження показали як можливість практичного застосування запропонованої технології на практиці, так і складність її реалізації: необхідність навчання одразу двох моделей, їх оптимізації, врахування відмінностей між різними наборами даних (якість знімків, масштаб, роздільна здатність, різні домени) тощо. Подальші дослідження будуть спрямовані на покращення навчання й донавчання як обраних, так і нових моделей, пошук й застосування підходів до оптимізації параметрів, а також розробки методів підвищення точності сегментації зруйнованих будівель як частини загальної інтелектуальної технології.

Список використаних джерел

1. Copernicus Emergency Management Service – Mapping. URL: https://mapping.emergency.copernicus.eu/ (дата звернення: 22.10.2026).

2. Ren S., He K., Girshick R., Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence. 2017*. Vol. 39, No. 6. P. 1137–1149. DOI: https://doi.org/10.1109/TPAMI.2016.2577031

3. Gupta R., Hosfelt R., Sajeev S., et al. xBD: A Dataset for Assessing Building Damage from Satellite Imagery. arXiv. 2019. arXiv:1911.09296. DOI: https://doi.org/10.48550/arXiv.1911.09296

УДК 004.92

Ірина Калініна
доктор технічних наук, професор,
професор кафедри
інтелектуальних інформаційних систем,
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Олександр Гожий
доктор технічних наук, професор,
професор кафедри
інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

СИСТЕМНИЙ ПІДХІД ДО РОЗВ'ЯЗАННЯ ЗАДАЧ КЛАСИФІКАЦІЇ З УРАХУВАННЯМ ДИСБАЛАНСУ КЛАСІВ

Актуальність. Сучасний рівень розвитку інформаційних технологій робить можливим застосування інтелектуальних методів для аналізу великих обсягів даних. Такі методи забезпечують автоматизацію процесів моніторингу, аналізу тенденцій та прогнозування під час вирішення різноманітних завдань.

Однак під час роботи з реальними наборами даних виникає низка проблем, найважливішою з яких є дисбаланс класів. У більшості випадків дані про критичні або аномальні ситуації представлені значно меншою кількістю спостережень порівняно зі звичайними умовами. Це призводить до того, що алгоритми машинного навчання (ML) здебільшого орієнтуються на більші обсяги даних і не здатні ефективно розпізнавати критичні випадки, які не можна ігнорувати і які мають першочергове значення при вирішенні прикладних завдань.

У зв'язку з цим особливого значення набуває розробка інтелектуальних систем (ІС), здатних розпізнавати дисбаланс класів і враховувати його під час вирішення завдань класифікації. Інтелектуальні інформаційні системи є одним із ключових напрямів сучасних інформаційних технологій, що дозволяють автоматизувати процеси аналізу, прогнозування та прийняття рішень. Вони ґрунтуються на використанні штучного інтелекту, машинного навчання, обробки даних і статистичного аналізу. Тому під час побудови інтелектуальних систем класифікації важливо систематично

застосовувати різноманітні методи та підходи, враховуючи обмеження, притаманні реальним даним. Такі системи повинні забезпечувати точне розпізнавання рідкісних і аномальних даних, що є запорукою своєчасного реагування та запобігання критичним ситуаціям. Особливо важливо при побудові таких систем враховувати особливості дисбалансу даних на кожному етапі вирішення завдання класифікації.

Метою цієї роботи є розробка систематичного підходу до вирішення задач класифікації з поетапним урахуванням дисбалансу класів.

Теоретична частина. Сукупність наявних методів і підходів до врахування дисбалансу класів у задачах класифікації можна об'єднати в кілька груп*: методи переформування вибірки (resampling); методи навчання з урахуванням функції втрат (loss-sensitive learning); методи оцінювання; ансамблеві методи; методи, що використовують додаткові дані.* Методи кожної групи призначені для застосування на окремих етапах розв'язання задачі машинного навчання.

У зв'язку з цим запропоновано схему систематичного підходу до врахування дисбалансу класів у задачах класифікації, яку наведено на рис. 1.

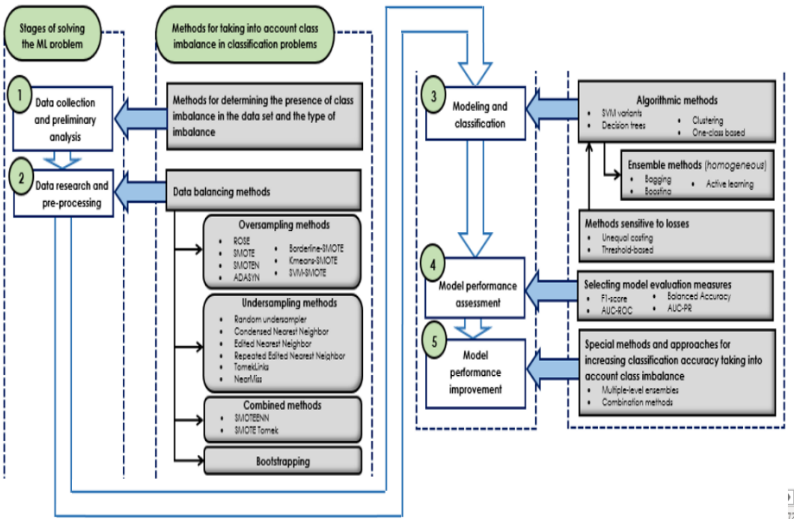


Рисунок 1. – Схема системного підходу до розв'язання задач класифікації з урахуванням дисбалансу класів.

У процесі розв’язання задач класифікації можна виокремити п’ять етапів: збір даних і їх попередній аналіз; дослідження та попередня обробка даних; моделювання і класифікація; оцінювання ефективності моделі; покращення результатів роботи моделі [1–3].

На кожному з цих етапів для врахування дисбалансу класів систематично застосовуються відповідні групи методів. На першому етапі використовуються методи для визначення наявності дисбалансу класів у наборі даних, а також його типу. На другому етапі – етапі попередньої обробки – безпосередньо застосовуються методи балансування даних (методи надлишкової вибірки, недостатньої вибірки, комбіновані методи або бутстрепінг). На етапі моделювання пропонується використовувати спеціальні алгоритми класифікації, які ефективно працюють за наявності дисбалансу в даних. Особливу увагу приділено ансамблевим методам і методам навчання з урахуванням функції втрат. На етапі оцінювання ефективності моделі застосовуються спеціальні метрики якості, адаптовані до умов дисбалансу даних. На п’ятому етапі пропонуються спеціальні методи та підходи для підвищення точності класифікації з урахуванням дисбалансу класів (багаторівневі ансамблі та комбіновані методи) [4–7].

Залежно від типу дисбалансу класів у конкретному наборі даних та характеристик цього набору даних, застосовуються різні комбінації методів із кожної групи.

Висновки. Представлено системний підхід до розв’язання задач класифікації з урахуванням дисбалансу класів. Розглянуто проблеми, пов’язані з дисбалансом класів, та проаналізовано його різновиди. Розглянуто основні концепції й методи вирішення проблеми дисбалансу класів. Розроблено покроковий системний підхід до розв’язання задач класифікації.

На основі цього підходу можливо створювати інтелектуальні системи класифікації, призначені для аналізу даних, їх попередньої обробки, балансування вибірок, а також побудови моделей машинного навчання для прогнозування. Застосування системного підходу та методів балансування є ключовими чинниками підвищення якості класифікації в реальних проектах, де спостерігається дисбаланс класів.

Список використаних джерел

1. Bidyuk P., Kalinina I., Gozhyj A., Pikh I, Chorna V., Gozhyi V. A Systematic Approach to Modeling and Forecasting Based on Real Data in Machine Learning Tasks / Zgurovsky, M., Pankratova, N. (eds.) System

Analysis and Data Mining . Studies in Systems, Decision and Control, 2025., Vol. 609. P. 71–87. Springer, Cham. DOI: https://doi.org/10.1007/978-3-031-97529-5_5

2. Gozhyj A., Kalinina I., Gozhyj V., Fuzzy Cognitive Analysis and Modeling of Water Quality. *The 9 th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications 21-23 September 2017, Bucharest, Romania.* DOI: 10.1109/IDAACS.2017.8095092

3. Babichev S., Liakh I., Kalinina I. Applying the Deep Learning Techniques to Solve Classification Tasks Using Gene Expression Data. *IEEE Access.* 2024. Vol. 12. P. (99):1-1. DOI: 10.1109/ACCESS.2024.3368070

4. He H., Garcia E. A. Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering.* 2009. Vol. 21, No. 9. P. 1263–1284. DOI: 0.1109/TKDE.2008.239

5. Grina F., Elouedi Z., Lefevre E. Learning from Imbalanced Data Using an Evidential Undersampling-Based Ensemble. *15th International Conference on Scalable Uncertainty Management, SUM'2022, Oct 2022, Paris, France.* P.235–248. DOI: 10.1007/978-3-031-18843-5_16, hal-03840612. URL: https://hal.science/hal-03840612v1

6. Madhura Prabha R., Sasikala S. Data Analytics for Imbalanced Dataset. *Journal of Computer Science.* 2024. Vol. 20, No. 2. P. 207–217. DOI: 10.3844/jcssp.2024.207.217

7. Delen D. Four Techniques to Handle Imbalanced Datasets. KNIME Blog. July 18, 2025. URL: https://www.knime.com/blog/4-techniques-handle-imbalanced-datasets

УДК 004.82

Олександр Ковалів
аспірант кафедри інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Галина Кондратенко
доктор технічних наук, професор,
професор кафедри інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євген Сіденко
кандидат технічних наук., доцент,
завідувач кафедри інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ОПТИМІЗАЦІЯ НЕЙРОМЕРЕЖЕВИХ МОДЕЛЕЙ ДЛЯ ПРОГНОЗУВАННЯ ПОШИРЕННЯ ІНФЕКЦІЙНИХ ЗАХВОРЮВАНЬ НА ОСНОВІ ВІДБОРУ ІНФОРМАТИВНИХ ОЗНАК

Прогнозування поширення інфекційних захворювань дозволяє своєчасно оцінювати епідеміологічні ризики та підтримувати прийняття управлінських , особливо, в умовах надзвичайних ситуацій та воєнних конфліктів, які спричиняють суттєві зміни в динаміці захворюваності. За таких умов часові ряди характеризуються нестаціонарністю, структурними змінами та нерегулярними коливаннями, що ускладнює їх прогнозування.

Одним із перспективних підходів до розв’язання цієї проблеми є застосування штучних нейронних мереж, здатних виявляти складні нелінійні залежності у часових даних. На точність прогнозування впливає архітектура нейронної мережі, параметри її навчання, а також структура вхідного простору. Використання недостатньо інформативних або надлишкових ознак може негативно впливати на якість прогнозів. Тому актуальним завданням є визначення набору вхідних змінних, який забезпечує підвищення точності прогнозування без додаткового ускладнення нейромережевої моделі.

Для проведення експериментальних досліджень використано

статистичні дані Центру громадського здоров'я Міністерства охорони здоров'я України щодо поширення 57 інфекційних захворювань за період із грудня 2016 року до січня 2024 року [1]. На основі отриманих даних сформовано часові ряди щомісячної захворюваності, виконано їх попереднє опрацювання та нормалізацію. На першому етапі проведено порівняльний аналіз шести нейромережевих архітектур: RNN, Block RNN, N-BEATS, N-HiTS, TCN та Transformer. Прогнозування здійснювалося на основі історичних даних за 24 місяці з горизонтом прогнозування 6 місяців. За результатами експериментів встановлено, що архітектура N-BEATS забезпечує найкращу точність серед досліджених моделей. Спільне навчання нейронної мережі на декількох часових рядах виявилося ефективнішим порівняно з навчанням на даних окремого захворювання [2]. Подальше дослідження було спрямоване на підвищення точності моделі N-BEATS шляхом включення до її вхідного простору екзогенних змінних. Як додаткові ознаки розглядалися календарний місяць z_1, сезон року z_2 та нормалізована чисельність населення z_3. Для визначення найбільш інформативного набору виконано перебір усіх можливих комбінацій зазначених змінних. При цьому архітектура нейронної мережі, параметри навчання та горизонт прогнозування залишалися незмінними. Результати оцінювання точності прогнозування за різних конфігурацій вхідного простору наведено в табл. 1.

Таблиця 1. – Вплив конфігурації вхідного простору на точність нейромережевого прогнозування

№	z_1	z_2	z_3	MAPE, %
1	–	–	–	82,92
2	+	–	–	79,10
3	–	+	–	80,35
4	–	–	+	79,80
5	+	+	–	77,95
6	+	–	+	73,56
7	–	+	+	78,40
8	+	+	+	78,41

Найкращі результати отримано для конфігурації, що включає календарний місяць та нормалізовану чисельність населення (варіант 6), MAPE = 73,56%, тоді як для базової моделі без використання екзогенних змінних – 82,92%. Відносне зменшення похибки становить

11,3%. Отримані результати підтверджують доцільність цілеспрямованого відбору інформативних змінних замість простого розширення вхідного простору нейронної мережі. Для візуального порівняння результатів прогнозування розглянуто динаміку захворюваності на грип із використанням базової моделі N-BEATS та моделі з оптимізованим вхідним простором (рис. 1).

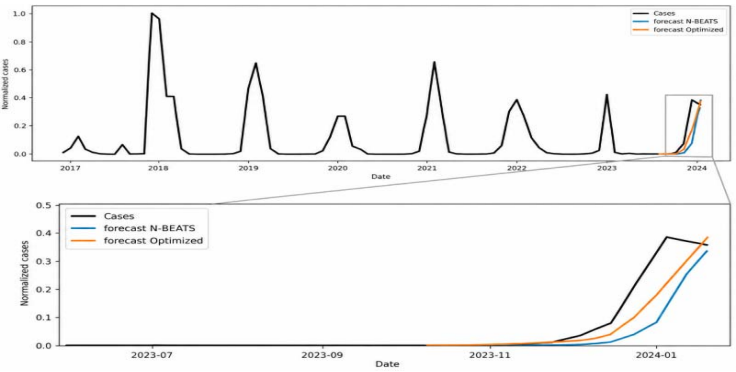


Рисунок 1. – Порівняння прогнозів захворюваності на грип за базовою та оптимізованою моделями N-BEATS

Обидві моделі відтворюють прогнозовану динаміку захворюваності, але використання оптимізованого набору вхідних ознак забезпечує уточнення прогнозних значень.

В результаті даної роботи, проведені дослідження показали, що оптимізація структури вхідних даних є ефективним способом підвищення точності нейромережевого прогнозування поширення інфекційних захворювань. Відбір інформативних екзогенних ознак дозволив покращити результати прогнозування без зміни архітектури N-BEATS та ускладнення процедури її навчання. Отримані результати можуть бути використані під час розроблення інтелектуальних систем моніторингу епідеміологічної ситуації та підтримки прийняття рішень у сфері охорони здоров'я.

Список використаних джерел

1. Public Health Center of the Ministry of Health of Ukraine. Infectious disease incidence in the population of Ukraine. https://phc.org.ua/kontrol-zakhvoryuvan/inshi-infekciyni-zakhvoryuvannya/infekciyna-zakhvoryuvanist-naselennya-ukraini (дата

звернення 26.10.2026).
2. Kovaliv, O., et al.: Improving the accuracy of infectious disease forecasts based on comparing neural network architectures. *Computation.* 2026. Vol. 14, No 2. Art. 54. DOI: 10.3390/computation14020054
УДК 004.942:519.17.85:621.396

Інесса Кулаковська
кандидат фіз.-мат. наук, доцент,
доцент кафедри інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

МАТЕМАТИЧНА МОДЕЛЬ НА ОСНОВІ МАТРОЇДІВ ДЛЯ ПІДВИЩЕННЯ СТІЙКОСТІ МОРСЬКИХ МЕРЕЖ ЗВ'ЯЗКУ ТА МОНІТОРИНГУ

У контексті **стійкості морських мереж зв'язку та моніторингу** (датчики забруднення води, метеостанції, буї цунамі, підводні гідроакустичні сенсори, супутникові термінали портів), матроїди є фундаментальною математичною базою. Морське середовище агресивне: шторми, автономність батарей та завади зв'язку постійно виводять датчики з ладу. Тому мережа повинна бути не просто дешевою, а **відмовостійкою (resilient)**.

Математична модель на основі матроїдів, які вирішують це завдання для сталого розвитку галузі: побудова k-зв'язних топологій через графічні матроїди. Для того, щоб мережа моніторингу не розпалася на ізольовані частини при відмові одного буя, вона повинна мати надлишкові канали зв'язку.

Нехай мережа – це граф $G = (V, E)$, де V – датчики, а E – можливі канали зв'язку. Класичний графічний матроїд шукає кістякове дерево (зв'язність $k = 1$). Для стійкості нам потрібна k-зв'язність (мережа витримує відмову $k - 1$ вузлів або ребер). Використовується теорема Неша – Вільямса про **покриття графа незалежними кістяковими деревами**. Ми шукаємо перетин або об'єднання кількох графічних матроїдів. Жадібний алгоритм будує k реберно-неперетинних кістякових дерев.

Якщо шторм знищить основну лінію передачі даних від океанічного буя до берега, інформація про екологічний стан миттєво

перенаправиться по одному з резервних дерев, які гарантовано не перетинаються за критичними вузлами.

Нижче наведено концептуальну схему оптимальної, стійкої структури зв'язку, де за допомогою графічного матроїда (рис. 1) виділено два абсолютно незалежні маршрути (два кістякові дерева) для передачі критичних екологічних даних від морської платформи до берегової станції. Навіть повне знищення одного з каналів не перериває моніторинг.

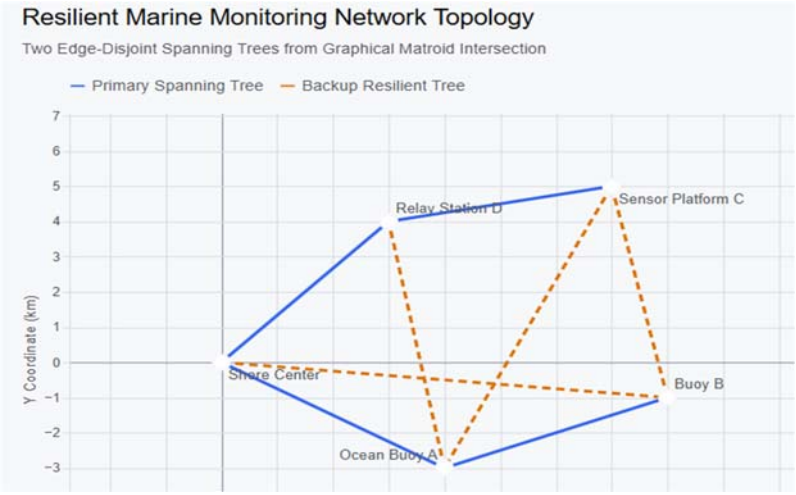


Рисунок 1. – Приклад топології мережі морського моніторингу

Розглянемо завдання для побудови **матриці лінійного матроїда** для кодування сигналів у морських мережах (Network Coding), яка захищає дані від завад і помилок. Застосування лінійних матроїдів у **мережевому кодуванні (Network Coding)** для морських мереж дозволяє вирішити проблему втрати пакетів через завади та шторми. Замість простої маршрутизації (пересилання копій), проміжні буї чи ретранслятори виконують лінійні комбінації отриманих пакетів над скінченним полем $Galois\ Field\ \mathbb{GF}(q)$.

Матриця лінійного матроїда в даній задачі є генераторною матрицею коду, де кожен стовпець відповідає певному каналу зв'язку або пакету. Властивість матроїда гарантує: якщо будь-які k стовпців є лінійно незалежними, то береговий центр зможе повністю відновити

початкові екологічні дані, навіть якщо частина пакетів буде втрачена через завади.

Математична постановка задачі. Припустимо, у нас є автономна морська платформа, яка генерує $k = 3$ критичні пакети даних (наприклад: p_1 – рівень радіації/нафтопродуктів, p_2 – температура води, p_3 – висота хвиль). Мережа передає дані через завадостійкий канал, використовуючи лінійний матроїд для кодування у векторний простір розмірності $n = 6$ (разом із надлишковими пакетами захисту). Кодування виконується над полем $\mathbb{GF}(2)$ або $\mathbb{GF}(2^m)$ для простоти обчислень.

Побудова матриці матроїда M

Для забезпечення максимальної відмовостійкості, аналог коду Ріда-Соломона, ми будуємо матрицю G розміром $k \times n$ (тобто 3×6). Будь-які 3 стовпці цієї матриці мають бути лінійно незалежними.

Ось матриця лінійного матроїда M над полем $\mathbb{GF}(5)$ (для ідеальної лінійної незалежності будь-якої трійки каналів):

$$G = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 2 & 3 \\ 0 & 0 & 1 & 1 & 4 & 2 \end{pmatrix}$$

Кожен стовпець $e_i (i = 1 \dots 6)$ відповідає пакету, який надсилається в морську мережу:

- e_1, e_2, e_3 — чисті інформаційні пакети (одинична підматриця);
- e_4, e_5, e_6 — закодовані лінійні комбінації для захисту від завад.

Морська платформа відправляє в ефір 6 сигналів (пакетів y_i), які формуються перемноженням вектора данних P на стовпці матриці матроїда:

$P = (p_1, p_2, p_3)^T$

1) $y_1 = p_1$;
2) $y_2 = p_2$;
3) $y_3 = p_3$;
4) $y_4 = p_1 + p_2 + p_3$;
5) $y_5 = p_1 + 2p_2 + 4p_3$;
6) $y_6 = p_1 + 3p_2 + 2p_3$.

(Усі операції додавання та множення виконуються за модулем 5).

Завдяки тому, що векторна система стовпців матриці G утворює **лінійний матроїд рангу 3**, виконуються такі умови.

– Базиси матроїда: будь-яка трійка стовпців $\{e_i, e_j, e_m\}$, яка є лінійно незалежною, утворює *базис матроїда*. Наприклад, набір $\{e_1, e_4, e_6\}$ має визначник (детермінант) не рівний нулю.

– Стійкість до втрат: якщо через шторм чи завади радіозв'язку пропадуть будь-які 3 пакети з 6 (наприклад, повністю затопило ретранслятор, що передавав y_1, y_2, y_3), берегова станція приймає пакети y_4, y_5, y_6.

Оскільки стовпці e_4, e_5, e_6 є лінійно незалежними в нашому матроїді, береговий центр вирішує просту систему лінійних рівнянь:

$$\begin{cases} y_4 = p_1 + p_2 + p_3 \\ y_5 = p_1 + 2p_2 + 4p_3 \\ y_6 = p_1 + 3p_2 + 2p_3 \end{cases}$$

І миттєво відновлює початкові екологічні показники p_1, p_2, p_3. Мережа демонструє **100% автономну регенерацію даних** без необхідності повторного запиту пакетів, який у морських умовах витрачає забагато енергії акумуляторів.

Висновок. Запропонована математична модель на основі матроїдів демонструє ефективність у забезпеченні стійкості морських мереж зв'язку та моніторингу в умовах агресивного середовища. Використання графічних матроїдів для побудови k-зв'язних топологій гарантує відмовостійкість мережі завдяки наявності незалежних кістякових дерев, що забезпечують резервні маршрути передачі даних. Паралельно застосування лінійних матроїдів у мережевому кодуванні дозволяє компенсувати втрати пакетів, спричинені завадами чи стихійними явищами, шляхом використання генераторних матриць із властивістю лінійної незалежності будь-якої підмножини каналів.

Такий підхід забезпечує повну реконструкцію критичних екологічних даних навіть при частковій втраті сигналів, що мінімізує енергетичні витрати та підвищує автономність системи. Отже, поєднання графічних і лінійних матроїдів формує теоретично обґрунтовану та практично реалізовану основу для створення відмовостійких, енергоефективних і довготривало функціональних морських мереж моніторингу, що відповідає вимогам сталого розвитку та безперервного екологічного контролю.

Список використаних джерел

1. Vorokhobin I., Zhuravska I., Burmaka I., Kulakovska I. Analysis of the error distribution density convergence with its orthogonal decomposition in navigation measurements. *Journal of Physics: Conference Series.* 2021. Vol. 2090, No. 1. Art. 012126. DOI: https://doi.org/10.1088/1742-6596/2090/1/012126
2. Kulakovska I. V. Інтелектуальна система планування в системах персонального тайм-менеджменту за допомогою матроїда : *матеріали XXVIII Всеукраїнської щорічної науково-практичної конференції «Могилянські читання – 2025: Досвід та тенденції розвитку суспільства в Україні: глобальний, національний та регіональний аспекти».* (10-14 листопада 2025 р., м. Миколаїв). Миколаїв : ЧНУ імені Петра Могили, 2025. С. 21–31.
3. Kulakovska I. Fundamental structures of combinatorics: Systematization of classes of matroids and their properties. *Modern Engineering and Innovative Technologies.* 2026. Vol. 2, No. 44-02. P. 292–301. DOI: https://doi.org/10.30890/2567-5273.2026-44-02-136

УДК 378(477)

Олександр Мещанінов
доктор педагогічних наук, професор,
професор кафедри
інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

СИНЕРГІЯ ЄДНОСТІ ВИКЛАДАЧІВ ЗІ СТУДЕНТАМИ УНІВЕРСИТЕТУ ІНФОРМАЦІЙНОГО СВІТУ

Вибуховий характер розвитку інформаційних технологій (ІТ) в усіх сферах діяльності людей XXI століття, людей інформаційного світу, потребує революційних змін університетської системи освіти. Поява та стрімке розповсюдження нового програмного забезпечення (ПЗ) не залишає достатнього проміжку часу на його опанування діючими науковими кадрами, викладачами університету. Нова генерація, як то кажуть, находу і оволодіває і застосовує новітні розробки ПЗ. Саме залучення молодого покоління захоплених та ініціативних розробників і є тим природним рішенням до розв'язку окресленої проблеми. Тобто,

новітня форма організації єдності викладачів зі студентами є природним послідовним етапом в еволюції університету [3,4]. Своєчасність такого єднання і є запорукою сумісного успіху. Гоноровість окремих професорів та адміністраторів лише обтяжує неухильність цього єднання.

Єднання викладачів зі студентами спостерігається під час вдосконалення, оновлення, розширення, тобто розвитку наявного навчального матеріалу через залучення нових форм представлення змісту, що передається інформацією. **Представлення змісту різноманіттям форм є сучасною методологією передавання знань**. Це і є глибоким, а не поверховим, рівнем оволодіння навчальним матеріалом і саме так відбувається оволодіння знаннями, опанування суттю питання.

Згадаємо слова Конфуція, які на сьогодні є кредом інноваційного навчання:

«Те, що я чую, я забуваю.
Те, що я бачу й чую, я трохи пам'ятаю.
Те, що я чую, бачу й обговорюю, я починаю розуміти.
Коли я чую, бачу, обговорюю й роблю, я набуваю знань і навичок.
Коли я передаю знання іншим, я стаю майстром».

А ми додамо: **Коли студент запроваджує, залучає нові форми представлення змісту, знання, він стає викладачем-творцем.** І не менше.

Отже, як визначив магістрант групи 501м Костюк Дмитро Олегович (2026) відбувається:

1. **«Еволюція форматів навчальних матеріалів** – перехід від друкованого тексту до гіпертексту, мультимедійного та інтерактивного контенту.
2. **Філософія відкритих освітніх ресурсів (OER)** – університет позиціонує себе не лише як споживача, а й як виробника відкритого контенту, наголошуючи на відповідальності ділитися філіппінським досвідом зі світом.
3. **Студенти як співтворці академічних текстів** – це, на мій погляд, найцінніша ідея виступу. Студенти не просто споживають матеріал, а самі знімають документальні відео, беруть інтерв'ю, досліджують культурну спадщину своїх спільнот.
4. **Міжнародна колаборація** – курс «Artin ASEAN Region» як приклад партнерства кількох відкритих університетів регіону для збереження культурної автентичності.

Вважаю підхід університету прогресивним і актуальним навіть зараз. Особливо імпонує ідея, що онлайн-навчання – це не просто заміна очного формату цифровим, а якісно нова модель взаємодії, де студент стає активним дослідником і автором контенту, а не пасивним отримувачем інформації» [1].

Також наведемо думки магістранта групи 501м Хорошева Руслана Андрійовича: «Цікавим прикладом був курс «Мистецтво в регіоні АСЕАН». Його створювали разом кілька відкритих університетів Південно-Східної Азії, а замість самих лише текстів використовували документальні фільми з інтерв'ю митців, архітекторів, кураторів і місцевих жителів. Авторка слушно зауважує, що письмове слово не завжди може передати справжній дух культури. Мені сподобалося, що студенти теж знімали власні документальні фільми про мистецтво своїх громад, а потім оцінювали роботи одне одного, обираючи три найкращі. Так вони вчаться не лише створювати, а й критично оцінювати.

Водночас хочу висловити кілька зауважень. Значну частину відео становлять уривки з документальних фільмів без чіткого пояснення, тому іноді важко зрозуміти, про що саме йдеться. Крім того, такий формат навчання потребує від студента багато часу, технічних навичок і доброго обладнання, а це доступно не всім. Також виникає питання, як оцінювати творчі мультимедійні роботи об'єктивно.

Загалом матеріал переконав мене, що відкрита дистанційна освіта може бути не просто вимушеною заміною аудиторних занять, а окремою філософією навчання. Вона робить освіту доступнішою, розвиває творче й критичне мислення та допомагає зберігати культурну спадщину. Я вважаю, що деякі з цих ідей, зокрема створення студентами власних мультимедійних проектів, можна було б успішно використовувати і в нашому університеті» [1].

Ось думки магістранта групи 501м Тимків Олександра Олександровича: «Це надзвичайно прогресивна і життєздатна модель сучасної освіти, яка відповідає на виклики цифрової епохи. Підхід, наведений у відео-матеріалі, демонструє глибоке розуміння того, що **технології мають слугувати не просто засобом доставки інформації, а інструментом для розширення можливостей**.

Особливо сильною є ідея відмови від ізольованого академічного середовища на користь публічності. Коли студенти знімають документальні фільми про свою громаду та викладають їх у відкритий доступ, вони не лише здобувають навички роботи з мультимедіа, а й займаються збереженням власної культурної спадщини. **Конструктивістський підхід, як навчання через створення, формує**

фахівців, які вміють критично мислити, досліджувати, взаємодіяти з різними культурами та адаптуватися до мінливого світу. Це чудовий приклад того, як онлайн-освіта може бути глибшою та більш залученою, ніж традиційні аудиторні лекції» [1].

А ось думки магістранта групи 501м Ареф'єва Олександра Сергійовича: «У відео Грейс Альфонсо розповідає, як Відкритий університет Філіппін навчає студентів онлайн. **Університет від початку зробив ставку на Інтернет: створює відкриті освітні ресурси, запускає безкоштовні онлайн-курси, а його студенти навчаються більш ніж у 40 країнах**. **Головна думка доповіді в тому, що студенти не просто читають матеріали, а самі їх створюють.** Наприклад, у курсі про мистецтво країн Азії вони знімають документальні фільми про місцевих митців, викладають їх на YouTube і оцінюють роботи одне одного.

Мені цей підхід дуже сподобався. Коли студент сам щось досліджує і створює, він запам'ятовує матеріал набагато краще, ніж після звичайної лекції. **До того ж, якщо твою роботу побачить багато людей, хочеться зробити її якісно, а не просто заради оцінки.** Також мені здається правильною думка, що університет має не лише користуватися чужими матеріалами, а й створювати власні. Українським вишам теж варто більше робити відкритих курсів українською мовою.

Водночас я бачу й деякі складнощі. Викладачу дуже важко перевірити сотні студентських відео та дати на кожне нормальний відгук. Крім того, не в кожного студента є хороша техніка і стабільний інтернет, а в Україні через відключення світла це особливо відчутно. Тому такі завдання варто робити гнучкими і давати можливість виконати їх в іншому форматі.

Загалом матеріал мені був цікавим і корисним. Він показує, що дистанційне навчання – це не просто лекції в Zoom, а можливість зробити навчання більш живим і практичним. На мою думку, деякі ідеї з цього досвіду, наприклад **проектні завдання та взаємне оцінювання**, цілком можна використовувати і в наших університетах» [1].

Цікаве висловлювання магістрантки групи 501м Врадій Вікторія Сергіївна: «Особливо мені сподобалася **ідея використання відкритих освітніх ресурсів**. Це навчальні матеріали, які доступні в інтернеті та можуть вільно використовуватися студентами й викладачами. Також цікавим є підхід, за якого студенти не просто отримують готову інформацію від викладача, а самі беруть участь у створенні нових

матеріалів. Наприклад, вони можуть створювати відео, проводити дослідження та обговорювати роботи своїх однокурсників.

Мені також сподобався приклад курсу про мистецтво країн Асоціації держав Південно-Східної Азії. Замість того щоб вивчати культуру лише за підручниками, студенти створюють документальні відео та показують справжні історії, думки й досвід людей зі своїх країн. На мою думку, такий спосіб навчання допомагає краще зрозуміти культурне різноманіття та побачити різні погляди на мистецтво.

Загалом, я вважаю, що дистанційне та відкрите навчання має багато переваг. Воно дозволяє людям навчатися незалежно від місця проживання та використовувати різні формати інформації: текст, відео, фотографії, обговорення та власні дослідження. **При цьому студент повинен бути активним, самостійно шукати інформацію, аналізувати її та використовувати отримані знання на практиці»** [1].

Наведемо думки магістранта групи 501м Гапішко Владислава Костянтиновича: «Після перегляду цього відео мені найбільше запам'яталася думка про те, що сучасна освіта не повинна зводитися лише до читання матеріалів і запам'ятовування інформації. Мені сподобалася ідея, що студент може бути не просто слухачем або споживачем знань, а активним учасником навчального процесу – досліджувати, створювати власні матеріали, обговорювати їх з іншими та фактично ставати співтворцем знань.

Також мені здалося цікавим те, наскільки важливу роль у такому навчанні відіграють цифрові технології та мультимедіа. Відео, подкасти, онлайн-форуми або документальні проекти можуть передавати інформацію зовсім інакше, ніж звичайний текст. **Особливо це помітно, коли йдеться про культуру, мистецтво та досвід людей, адже іноді побачити або почути людину набагато цінніше, ніж просто прочитати її слова в підручнику**.

На мою думку, такий підхід робить навчання більш практичним і осмисленим. **Людина вчиться не тільки отримувати готову інформацію, але й самостійно її шукати, аналізувати, перевіряти, формулювати власну думку та представляти результати іншим.** Саме ці навички, як підкреслюється наприкінці відео, особливо важливі в сучасному цифровому світі.

Загалом після цього відео я ще раз задумався над тим, що технології в освіті – це не просто можливість перенести звичайну лекцію в онлайн. Вони можуть змінити сам підхід до навчання і дати

студентам значно більше свободи та відповідальності за те, як вони отримують і створюють знання» [1].

Таким чином, вочевидь наявна синергія єдності викладачів зі студентами університету інформаційного світу, спільність ставлення магістрантів до творчості, творчості що зароджується під час виконання завдань, у конкретному випадку, по нормативній дисципліни «Педагогіка вищої школи».

Як висновок, це висловлювання Марка Цукерберга: **«Мета - це те, що створює справжнє щастя»** [2].

Список використаних джерел

1. Власна думка магістрантів 501м та 571м груп факультету комп'ютерних наук Чорноморського національного університету (ЧНУ) імені Петра Могили щодо ознайомлення із запропонованим матеріалом із дисципліни «Педагогіка вищої школи». 2026. URL: https://moodlefcs.chmnu.edu.ua/mod/assign/view.php?id=1048&action=grading (дата звернення: 26.10.2026).

2. Марк Цукерберг: «Мета – це те, що створює справжнє щастя». URL: https://web.archive.org/web/20120428025409/http://moodle.chdu.edu.ua/ (дата звернення: 26.10.2026).

3. Мещанінов О. П., Боровльова С. Ю., Дворецька С. В. Синергія у єдності дистанційних форм навчання і наукових досліджень для сталого розвитку університетської системи освіти. *Наукові праці : Науково-методичний журнал. Серія : Педагогічні науки.* 2018. Том 313, Вип. 301. С. 10–14. URL: http://lib.chdu.edu.ua/pdf/naukpraci/pedagogika/2018/313-301-1.pdf (дата звернення: 26.10.2026).

4. Мещанінов О. П. Єдність дистанційних форм навчання і наукових досліджень: розширення можливостей сучасної університетської системи освіти. *Наукові праці : Науково-методичний журнал. Серія : Педагогічні науки.* 2016. Том 270, Вип. 258. С. 7–11. URL: http://lib.chdu.edu.ua/pdf/naukpraci/pedagogika/2016/270-258-1.pdf (дата звернення: 26.10.2026).

УДК 004.85

В'ячеслав Перепелиця
здобувач другого (магістерського) рівня вищої освіти
Чорноморський національний університет
ім. Петра Могили
м. Миколаїв, Україна
Олексій Козлов
доктор технічних наук, професор, професор
кафедри інтелектуальних інформаційних систем
Чорноморський національний університет
ім. Петра Могили
м. Миколаїв, Україна

ТЕХНОЛОГІЯ ОЦІНЮВАННЯ ЯКОСТІ ВИННОЇ ПРОДУКЦІЇ НА ОСНОВІ МЕТОДІВ ПРОСТОРОВОЇ КЛАСТЕРИЗАЦІЇ ТА АНСАМБЛЕВОГО НАВЧАННЯ

У даній роботі розглядаються питання комп'ютеризованого оцінювання якості виноробної продукції за її фізико-хімічними параметрами для задач автоматизованої кваліметрії та ідентифікації елітних вин. Подано огляд класичних алгоритмів машинного навчання і сучасних гібридних підходів на основі просторової кластеризації (SOM), синтетичного балансування (SMOTE) та градієнтного бустингу (XGBoost), що дають змогу подолати проблему дисбалансу класів і суттєво підвищити чутливість системи до розпізнавання високоякісної продукції у складних нелінійних середовищах.

В енології виділяють два основні підходи до оцінювання якості винної продукції. Перший – сенсорний (дегустаційний) аналіз, на противагу якому, існує базова лабораторна експертиза [1]. В основі першого способу лежить виключно суб'єктивний аналіз продукції спираючись на смакові рецептори спеціально навченої людини – дегустатора. Зрозуміло, що рецептори експерта не є відкаліброваним сенсором, та, в залежності від особистісних вподобань людини та дії зовнішніх впливів, можуть бути скомпрометовані. Це руйнує саму суть об'єктивного ціноутворення виноробної продукції. Другий спосіб ґрунтується виключно на фізико-хімічних властивостях вина. Безпристрасна лабораторна експертиза може розкласти досліджуваний напій на молекули, вимірявши щільність, кислотність, концентрацію сульфатів та інші складові елементів вина з точністю до тисячних

часток. Проте закономірності формування органолептичних властивостей винної продукції під впливом його фізико-хімічного складу є надзвичайно складним та нелінійним.

Тому існує потреба у створенні об'єктивного класифікатора, завдання якого – визначити якість вина за його фізико-хімічним складом спираючись лише на математичні алгоритми.

Використання класичних алгоритмів машинного навчання [2] (логістичної регресії, дерев рішень та ін.) для вирішення цього завдання стикається з ключовою проблемою – парадоксом точності. Він виникає через екстремальний дисбаланс класів у реальних виробничих умовах. Наприклад, у відомому масиві даних Vinho Verde [3], зібраному дослідниками на виноробнях португальської провінції Мінью у період з травня 2004 по лютий 2007 року, абсолютна більшість виготовленої продукції отримує оцінки 5-6 балів з 10, що відповідають вину звичайної якості. Тоді як елітні купажі із оцінками більше 7 балів становлять статистичну меншість. За подібних умов звичайні класифікатори мінімізують глобальну похибку за рахунок ігнорування міноритарного класу елітного вина, хибно класифікуючи його як звичайне.

Для вирішення цієї проблеми було спроєктовано багатоступеневий інженерний конвеєр (SMOTE-SOM-XGBoost), який послідовно застосовує методи балансування, топологічного екстрагування ознак та градієнтного бустингу (рис. 1).



Рисунок 1. – Загальний вигляд конвеєра машинного навчання (SMOTE-SOM-XGBoost) для оцінювання якості вина

Спочатку для 11 фізико-хімічних дескрипторів вилучаються шумові складові, які не мають значущої кореляції з цільовою сенсорною оцінкою. Для того, щоб усунути дисбаланс вибірки до тренувального набору даних, застосовано алгоритм Synthetic Minority Over-sampling Technique (SMOTE). Він не просто створює дублі записів елітного вина, а генерує унікальні синтетичні зразки використовуючи метод інтерполяції у багатовимірному просторі ознак між найближчими сусідами [4].

На другому етапі збалансований простір передається до

самоорганізовуваних карт Кохонена – Self-Organizing Map (SOM). Цей метод здійснює нелінійне кластеризування, відображаючи складні фізико-хімічні патерни на двовимірну технологічну решітку [5]. В результаті чого кожний зразок вина отримує по дві нові просторові координати, що вказують на його приналежність до специфічних макро- та мікрокластерів якості.

Фінальна класифікація формується за допомогою ансамблю XGBoost [6], який навчається на просторі збагаченому попереднім етапом. Алгоритм використовує розклад Тейлора другого порядку для оптимізації логарифмічної функції втрат, а вбудовані L_1/L_2 регуляризатори запобігають перенавчанню на синтезованих шумах.

Програмна реалізація гібридної технології виконана мовою Python [7] із використанням бібліотек scikit-learn [8], imblearn [9] та MiniSom [10]. Експеримент проводився на незалежній тестовій вибірці білого вина (25% від загального датасету), яка не піддавалася синтетичному балансуванню, що імітує реальні промислові умови з природним дефіцитом елітної продукції.

Отримані результати достовірності класифікації якості білого вина за допомогою розглянутих методів показані в таблиці 1.

Таблиця 1. – Результати класифікації якості білого вина

Метод класифікації	**Кількість зразків у навчальній вибірці, шт.**	**Кількість зразків у тестовій вибірці, шт.**	**Достовір ність класи фікації на навчальній вибірці, %**	**Достовірність класифікації на тестовій вибірці, %**
Метод логістичної регресії	3673	1225	80,5	81,2
Метод машини опорних векторів (SVM)	3673	1225	83,1	82,4
Метод дерев рішень (CART)	3673	1225	84,0	80,7
Запропонований підхід (SOM+SMOTE+ XGBoost)	3673	1225	98,2	83,8

Із таблиці 1 видно, що запропонований підхід показує найвищу достовірність на навчальній вибірці (близько 98,2% завдяки потужності градієнтного бустингу XGBoost та збагаченню топологічними ознаками) і демонструє найкращий результат

узагальнення на незалежній тестовій вибірці (83,8%).

Аналіз ефективності розробленого алгоритму у порівнянні з традиційними методами машинного навчання показано у таблиці 2.

Таблиця 2 – Аналіз ефективності алгоритмів

Показник ефективності / Метод класифікації	**Accuracy**	**Precision**	**Recall**	**F1-Score**
Метод логістичної регресії	0,8122	0,6522	0,2830	0,3947
Метод машини опорних векторів (SVM)	0,8237	0,7475	0,2972	0,4066
Метод дерев рішень (CART)	0,8073	0,6058	0,3132	0,4129
Запропонований підхід (SOM+SMOTE+XGBoost)	0,8384	0,6227	0,6415	0,6320

Аналіз базових моделей підтвердив їхню вразливість до дисбалансу. При загальній точності (Accuracy) на рівні 80-82%, їхня здатність виявляти елітне біле вино (Recall) склала близько 28-31%. Це означає, що алгоритми пропускають близько 70% високоякісних напоїв, що є неприпустимо для промислових виробників.

Застосування розробленої гібридної архітектури (SOM + SMOTE + XGBoost) кардинально змінило структуру розпізнавання. Показник повноти виявлення елітного класу (Recall) зріс до 64,15%, що свідчить про успішне подолання моделлю парадоксу точності. Комплексна метрика F1-Score збільшилася до 63,2%, а показник влучності (Precision) хоча і програє деяким конкурентам, проте залишається на прийнятному рівні 62,2%.

Отже, впровадження інформаційної технології на основі запропонованого підходу, який поєднує в собі методи просторової кластеризації (SOM) та синтетичного балансування (SMOTE) як етапи передобробки для ансамблю XGBoost, дозволяє вирішити проблему інформаційного зашумлення та дисбалансу класів у задачах енологічної кваліметрії. Експериментально доведено, що збагачення простору хімічних дескрипторів топологічними координатами підвищує точність ідентифікації елітного білого вина у кілька разів порівняно з класичними статистичними класифікаторами.

Список використаних джерел

1. Cortez P., Cerdeira A., Almeida F., Matos T., Reis J. Modeling wine preferences by data mining from physicochemical properties. Decision Support Systems. 2009. Vol. 47, No. 4. P. 547–553. DOI: https://doi.org/10.1016/j.dss.2009.05.016
2. Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd ed. Springer, 2009. 745 p. URL: https://hastie.su.domains/ElemStatLearn/ (дата звернення: 26.10.2026).
3. Wine Quality. URL: https://archive.ics.uci.edu/dataset/186/wine+quality (дата звернення: 26.10.2026).
4. Chawla N. V., Bowyer K. W., Hall L. O., Kegelmeyer W. P. SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research. 2002. Vol. 16. P. 321–357. DOI: https://doi.org/10.1613/jair.953
5. Kohonen T. Self-Organizing Maps. 3rd ed. Springer, 2001. 501 p. DOI: https://doi.org/10.1007/978-3-642-56927-2
6. Chen T., Guestrin C. XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016. P. 785–794. DOI: https://doi.org/10.1145/2939672.2939785
7. Python Software Foundation. Python Language Reference, Version 3.11. URL: https://docs.python.org/3.11/ (дата звернення: 26.10.2026).
8. Pedregosa F., Varoquaux G., Gramfort A., et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. 2011. Vol. 12. P. 2825–2830. URL: https://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html (дата звернення: 26.10.2026).
9. Lemaître G., Nogueira F., Aridas C. K. Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. Journal of Machine Learning Research. 2017. Vol. 18, No. 17. P. 1–5. URL: https://jmlr.csail.mit.edu/papers/v18/16-365.html (дата звернення: 26.10.2026).
10. Vettigli G. MiniSom: minimalistic and NumPy-based implementation of the Self Organizing Map. GitHub, 2018. URL: https://github.com/JustGlowing/minisom (дата звернення: 26.10.2026).

УДК 004.942:519.17

Ганна Петрина
здобувачка першого (бакалаврського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Віктор Гожий
кандидат технічних наук,
старший викладач кафедри
інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ІНФОРМАЦІЙНА СИСТЕМА БАГАТОКАНАЛЬНОЇ АТРИБУЦІЇ ТА ОПТИМІЗАЦІЇ МАРКЕТИНГОВИХ БЮДЖЕТІВ НА ОСНОВІ ГРАФОВИХ МОДЕЛЕЙ

Актуальність теми. У сучасному цифровому маркетингу шлях клієнта до цільової дії формується з послідовності різнотипних контактів – показів медійної реклами, переходів із соціальних мереж, відвідувань блогу, ретаргетингу та пошукової реклами [1-4]. Класичні евристичні моделі атрибуції (Last-Click, First-Click, Linear) штучно спотворюють оцінку внеску каналів у конверсію: правило останнього кліку завищує цінність каналів «закриття угоди» та знецінює канали першого знайомства з брендом. Промислові «чорноскринькові» рішення (Data-Driven Attribution у Google Analytics 4 тощо) є непрозорими, не піддаються аудиту та не підтримують прогнозне сценарне моделювання. Це створює потребу у відкритому алгоритмічному інструментарії, що поєднує справедливий розподіл заслуг між каналами з можливістю оптимізації майбутнього маркетингового бюджету [5,6].

Мета роботи. Розробити відкриту інтерпретовану інформаційну систему, яка на основі ймовірнісних (ланцюги Маркова) та теоретико-ігрових (вектор Шеплі) графових моделей здійснює багатоканальну атрибуцію внеску каналів у воронку продажів і забезпечує сценарне моделювання («What-If» аналіз) розподілу маркетингового бюджету з урахуванням кривих насичення каналів.

Принцип роботи. Нехай задано множину маркетингових каналів $C = \{c_1, c_2, \ldots, c_n\}$ та набір користувацьких шляхів P, де кожен шлях $p_i \in$

P є впорядкованою послідовністю контактів з каналами, що завершується станом «конверсія» або «відмова». На цих даних здійснюються такі дії (рис. 1):

– побудова стохастичної матриці переходів $\Pi = [\pi_{ij}]$ між станами орієнтованого графу $G = (V, E)$;

– обчислення ваги атрибуції w_k двома незалежними методами;

– побудова функції відгуку каналів $f_k(x_k)$ із властивістю спадної віддачі;

– розв'язання задачі умовної оптимізації: $\max \Sigma f_k(x_k)$ за умови $\Sigma x_k = B$, $x_k \geq 0$, де B — загальний бюджет.

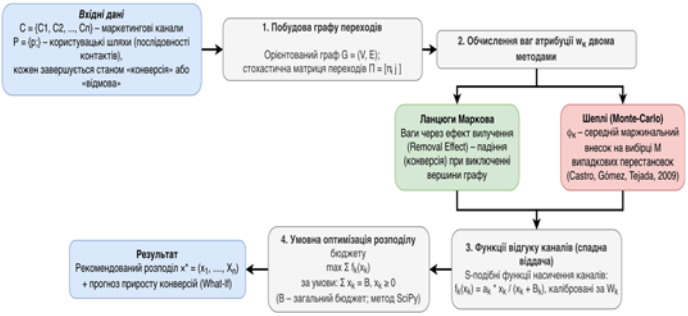


Рисунок 1. – Принцип роботи інформаційної системи атрибуції

Алгоритмічна база. В основу аналітичного ядра покладено два незалежні підходи до оцінювання внеску каналів:

– ймовірнісна модель на базі ланцюгів Маркова першого порядку: ваги атрибуції визначаються через ефект вилучення (Removal Effect) – відносне падіння ймовірності досягнення стану «конверсія» при виключенні відповідної вершини графу переходів;

– теоретико-ігрова модель на базі вектора Шеплі з Monte-Carlo-апроксимацією (Castro, Gómez, Tejada, 2009): канали трактуються як учасники кооперативної гри, а внесок кожного каналу φ_k оцінюється як середній маржинальний приріст характеристичної функції v(S) на вибірці з M випадкових перестановок каналів, що забезпечує обчислювальну складність $O(M \cdot n)$ замість $O(2^n)$ для точного комбінаторного перебору коаліцій.

Для модуля сценарного моделювання побудовано S-подібні функції насичення каналів виду $f_k(x_k) = \alpha_k \cdot x_k / (x_k + \beta_k)$, на яких розв'язується задача умовної опуклої оптимізації розподілу бюджету методами чисельної оптимізації бібліотеки SciPy [7].

Архітектура програмної реалізації. Систему спроєктовано як трирівневий вебзастосунок:

– аналітичне ядро мовою Python: препроцесинг і session stitching сирих логів клікстріму (pandas, polars), побудова графу переходів і стохастичних матриць (NetworkX), чисельна оптимізація розподілу ресурсів (SciPy);

– backend-рівень на FastAPI з REST-ендпоінтами для завантаження CSV-логів, запуску обчислювальних пайплайнів та експорту аналітичних звітів;

– frontend-дашборд на React / TypeScript / Tailwind CSS з інтерактивною візуалізацією переходів у вигляді Sankey-діаграми та графу мережі, порівняльними діаграмами часток каналів за різними моделями й What-If симулятором із повзунками коригування бюджету та миттєвим перерахунком прогнозованого приросту конверсій.

Дослідницька складова та експеримент. Передбачено проведення серії обчислювальних експериментів на датасеті клікстрім-подій: зіставлення евристичних моделей (Last-Click, First-Click, Linear) з алгоритмічними (Markov, Shapley) за часткою атрибутованих конверсій і стійкістю до зашумлення; експериментальне дослідження швидкості збіжності Monte-Carlo-оцінки Шеплі до істинного значення ($\sim 1/\sqrt{M}$) з побудовою довірчих інтервалів атрибуційних ваг та верифікацією на задачах малої розмірності проти точного комбінаторного розв'язку; аналіз чутливості результатів оптимізації до параметрів кривих насичення каналів.

Практичне застосування. Розроблена інформаційна система орієнтована на застосування:

– у аналітичних підрозділах eCommerce-компаній та digital-агентств;

– у задачах прозорого аудиту ефективності рекламних каналів;

– у сценарному плануванні перерозподілу маркетингового бюджету без прив'язки до закритих хмарних сервісів.

Висновки. Запропонована інформаційна система поєднує дискретно-ймовірнісний та теоретико-ігровий підходи до аналізу ланцюжків дотиків із модулем сценарного моделювання бюджету, формуючи відкриту й інтерпретовану алгоритмічну альтернативу комерційним «чорноскриньковим» платформам атрибуції. Перспективним напрямом подальших досліджень є розширення аналітичного ядра моделями глибокого навчання послідовностей (LSTM, архітектури з механізмом Self-Attention) та інтеграція методів пояснюваного штучного інтелекту (Explainable AI: Attention Weights,

DeepSHAP) для динамічного врахування повної історії клієнта та точних часових інтервалів між контактами.

Список використаних джерел

1. Shao X., Li L. Data-driven multi-touch attribution models. Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '11). ACM, 2011. P. 258–264. DOI: https://doi.org/10.1145/2020408.2020453
2. Anderl E., Becker I., von Wangenheim F., Schumann J. H. Mapping the customer journey: Lessons learned from graph-based online attribution modeling. *International Journal of Research in Marketing.* 2016. Vol. 33, No. 3. P. 457–474. URL: https://www.sciencedirect.com/science/article/pii/S0167811616000094 (дата звернення: 14.10.2026).
3. Shapley L. S. A value for n-person games. Contributions to the Theory of Games. Princeton University Press, 1953. Vol. 2. P. 307–317.
4. Zhao K., Mahboobi S. H., Bagheri S. R. Shapley value methods for attribution modeling in online advertising. arXiv. 2018. arXiv:1804.05327. URL: https://arxiv.org/abs/1804.05327 (дата звернення: 14.10.2026).
5. Dalessandro B., Perlich C., Stitelman O., Provost F. Causally motivated attribution for online advertising. Proceedings of the Sixth International Workshop on Data Mining for Online Advertising (ADKDD '12). ACM, 2012. DOI: https://doi.org/10.1145/2351356.2351363
6. ChannelAttribution: Markov Model for the Online Multi-Channel Attribution Problem. The Comprehensive R Archive Network (CRAN). URL: https://cran.r-project.org/package=ChannelAttribution (дата звернення: 14.10.2026).
7. Hagberg A., Schult D., Swart P. NetworkX documentation. URL: https://networkx.org/documentation/stable/ (дата звернення: 14.10.2026).

УДК 004.4:502.7

Олег Піонткевич
здобувач другого (магістерського) рівня вищої освіти
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Олексій Козлов
доктор технічних наук, професор,
професор кафедри
інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

АЛГОРИТМІЧНІ ЗАСАДИ ФОРМУВАННЯ ГІБРИДНОГО НАБОРУ ГЕОДАНИХ ОБ'ЄКТІВ ЗБОРУ ТВЕРДИХ ПОБУТОВИХ ВІДХОДІВ

У даній роботі запропоновано підхід до формування гібридного набору геоданих об'єктів збору твердих побутових відходів (ТПВ) шляхом доповнення картографованих даних OpenStreetMap синтетичними точками. Розглянуто оцінювання просторової інтенсивності розміщення об'єктів, генерацію та просторову фільтрацію синтетичних точок. На прикладі м. Вінниці сформовано демонстраційний набір із 188 записів, зокрема 88 картографованих об'єктів і 100 синтетичних точок. Отримані результати можуть використовуватися для імітаційного моделювання за умов неповноти відкритих картографічних даних.

Ефективне поводження з ТПВ передбачає своєчасне збирання, сортування [1] та транспортування відходів для подальшої переробки або утилізації. Водночас повторне використання складових ТПВ і виробництво вторинних матеріалів є важливими напрямами ресурсозбереження [2, 3]. Можливості корисного застосування різних промислових відходів і відпрацьованих матеріалів також досліджено в [4, 5].

Організація збирання ТПВ пов'язана не лише з екологічними, а й із транспортно-технологічними завданнями. У сміттєвозах застосовують механізоване завантаження та ущільнення відходів [6]. Для спеціалізованого обладнання запропоновано також методику розрахунку систем зневоднення ТПВ [7]. Ефективність використання

такого транспорту залежить, зокрема, від обґрунтованого планування послідовності обслуговування місць збору відходів.

Відтак, поєднання екологічних і транспортних завдань актуалізує розроблення геоданих засобів підтримки диспетчеризації сміттєвозів у межах концепції Smart City. Необхідною передумовою є впорядкований набір координат і характеристик місць збору ТПВ. Проте відкриті картографічні джерела, зокрема OpenStreetMap, можуть містити неповні відомості про такі об'єкти та їхні атрибути. Під час просторового моделювання доцільно враховувати близькість до проїзних доріг, особливості забудови й розміщення наявних об'єктів. Дорожню мережу доцільно представляти орієнтованим графом, вузли якого відповідають перехрестям і транспортним розв'язкам, а ребра – ділянкам доріг з урахуванням напрямів руху та обмежень проїзду [8].

Метою даної роботи є розроблення відтворюваного алгоритму формування гібридного набору геоданих об'єктів збору ТПВ із розмежуванням на картографовані і синтетичні. Алгоритм передбачає імпорт координат з OpenStreetMap, їх очищення, оцінювання просторової інтенсивності, генерацію додаткових точок-кандидатів, відхилення надто близько розташованих точок та інтеграцію результатів до спільного набору геоданих. За наявності дорожніх даних передбачено додаткову перевірку близькості точок-кандидатів до придатних для проїзду ділянок.

Математичною основою просторового моделювання є неоднорідний процес Пуассона, для якого ймовірність появи k точок у досліджуваній області S визначається за формулою:

$$P(N(S) = k) = \frac{(\Lambda(S))^k e^{-\Lambda(S)}}{k!}, \quad (1)$$

$$\Lambda(S) = \int_S \lambda(x) dS, \quad (2)$$

де $N(S)$ – кількість точок в області S, k = 0, 1, 2, … – можливе значення їхньої кількості, $\Lambda(S)$ – математичне сподівання $N(S)$, $\lambda(x)$ – просторова інтенсивність розміщення точок, x – довільна точка території.

Для демонстраційної реалізації невідому $\lambda(x)$ наближено ядерною оцінкою за наявними n точками OpenStreetMap [9]:

$$\hat{\lambda}_h(x) = \frac{1}{2\pi h^2} \sum_{i=1}^{n} \exp\left(-\frac{\|x - x_i\|^2}{2h^2}\right), \quad (3)$$

де h – параметр згладжування, n – кількість картографованих об'єктів, x_i – координати i-го об'єкта з OpenStreetMap.

За фіксованої кількості N нових точок-кандидатів їхні координати вибирають із щільністю, пропорційною оцінці $\hat{\lambda}_h(x)$ у межах області моделювання S. Після генерації виконують просторове відсіювання: перевіряють належність області S та мінімальну відстань до наявних і вже прийнятих точок. За доступності перевірених дорожніх геоданих додатково застосовують умову близькості до дороги. За фіксованого N і просторових обмежень реалізована генерація є умовним імітаційним алгоритмом, а не безпосередньою реалізацією неоднорідного процесу Пуассона:

$$d\left(y_j, L_{road}\right) \le D_{max}, \quad (4)$$

$$d\left(y_j, X_{obs} \cup Y_{gen}^{(j-1)}\right) \ge r_{min}, \quad (5)$$

де y_j – координати j-ї згенерованої точки-кандидата; L_{road} – множина придатних для проїзду ділянок доріг; D_{max} – максимально допустима відстань до дороги; X_{obs} – множина координат об'єктів із OpenStreetMap; $Y_{gen}^{(j-1)}$ – множина раніше згенерованих і прийнятих точок-кандидатів; r_{min} – мінімально допустима відстань між точками; $d(y_i, A)$ – найкоротша відстань від точки y_j до множини A. Значення D_{max} і r_{min} є параметрами дослідження, а не нормативами роботи сміттєвоза. Фактична транспортна доступність потребує окремої перевірки.

Для уточнення просторового положення синтетичних об'єктів запропоновано використовувати алгоритм прив'язки координат до дорожньої мережі Snap to Road на основі сервісу OSRM Nearest. За цим підходом згенерована точка проєктується на найближчу ділянку автомобільної дороги з контролем максимально допустимої відстані переміщення. Точки-кандидати, що не задовольняють установлених просторових обмежень, відхиляються.

Запропоновано зберігати для кожного об'єкта координати, ідентифікатор, джерело, статус перевірки, час імпорту, а за наявності – місткість, тип відходів та рівень заповненості. Реалізацію такого сховища можна здійснити засобами PostgreSQL/PostGIS із

використанням просторових індексів GiST і функцій перевірки відстаней.

Результати застосування запропонованого алгоритму формування геоданих інфраструктури збору ТПВ у м. Вінниці наведено на рис. 1. На основі 88 об'єктів, отриманих із відкритої картографічної бази OpenStreetMap за допомогою сервісу Overpass Turbo, сформовано додаткові 100 синтетичних точок із використанням методу ядерного оцінювання просторової щільності. Отриманий розподіл відображає просторові закономірності розміщення наявних об'єктів та дозволяє розширити вихідний набір геоданих до 188 записів двох категорій. Розподіл синтетичних точок відтворює тенденцію їх концентрації поблизу наявних картографованих об'єктів, що відповідає прийнятому способу оцінювання інтенсивності. Водночас згенеровані координати не засвідчують існування відповідних контейнерів; їхню придатність до розміщення, зокрема транспортну доступність, не підтверджено незалежною перевіркою.

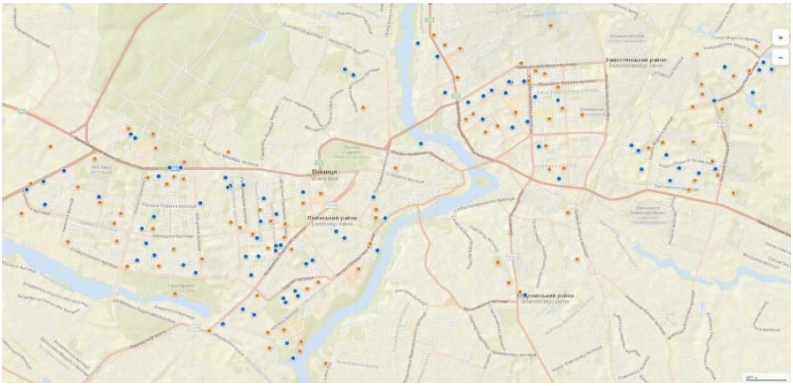


Рисунок 1. – Просторове розміщення об'єктів збору ТПВ у м. Вінниці за даними OpenStreetMap (сині) та результатами моделювання (помаранчеві)

Отримані результати дають змогу технічно сформувати гібридний набір геоданих для сценарних експериментів, але не характеризують повноту фактичної інфраструктури. Подальші дослідження передбачають залучення щільності житлової забудови, транспортної доступності, незалежних перевірених координат і побудову топологічного графа доріг. Лише після отримання маршрутних відстаней і підтверджених обсягів ТПВ можливе обґрунтоване

розв’язання задачі маршрутизації сміттєвозів із обмеженням вантажомісткості та часовими вікнами. Основою для таких подальших досліджень можуть стати підходи до підтримки прийняття рішень у транспортній логістиці [10] та оптимізації складних систем [11].

Список використаних джерел

1. Ahmad G., Aleem F. M., Alyas T., Abbas Q., Nawaz W., Ghazal T. M., *et al.* Intelligent waste sorting for urban sustainability using deep learning. *Scientific reports*. 2025. 15(1). 27078. 19 p. DOI: https://doi.org/10.1038/s41598-025-08461-w
2. Гордієнко О. А., Ранський А. П., Сандул О. М. Комплексна технологія перероблення хімічних складових різнопланових відходів та отримання пластичних мастил. *Сучасні технології, матеріали і конструкції в будівництві.* 2026. Вип. 40(1). С. 240–246. DOI: https://doi.org/10.31649/2311-1429-2026-1-240-246
3. Березюк О. В., Лемешев М. С., Стаднійчук М. Ю. Поширеність переробки золи при виготовленні будівельних матеріалів. Сучасні технології, матеріали і конструкції у будівництві.2023. № 2. 2023. С. 56–61. DOI: https://doi.org/10.31649/2311-1429-2023-2-56-61
4. Sandul O., Titov T., Kulyk A., Sakalova H., Shevchyk K., Petrushka K. Reuse of the spent sorbent mixture for wastewater treatment. Environmental problems. 2024. Vol. 9, No 4: P. 193–198. DOI: https://doi.org/10.23939/ep2024.04.193
5. Ranskiy A., Sandul O., Gordienko O., Didenko N., Titov T. Development of new C, S, N-containing plastic lubricants based on products from industrial waste integrated processing. Eastern-European Journal of Enterprise Technologies. 2024. Vol. 1, No 6. 2024. P. 13–21. DOI: https://doi.org/10.15587/1729-4061.2024.296622
6. Березюк О. В. Моделювання компресійної характеристики твердих побутових відходів у сміттєвозі на основі комп’ютерної програми “PlanExp”. *Вісник Вінницького політехнічного інституту.* 2016. № 6. С. 23–28. URL: https://visnyk.vntu.edu.ua/index.php/visnyk/article/view/1982
7. Березюк О. В. Методика інженерних розрахунків параметрів обладнання для зневоднення твердих побутових відходів у сміттєвозі. *Вісник Вінницького політехнічного інституту.* 2020. № 2. С. 73–81. DOI: https://doi.org/10.31649/1997-9266-2020-149-2-73-81
8. Boeing G. Modeling and analyzing urban networks and amenities with OSMnx. Geographical Analysis, 57(4), 2025. P. 567–577. DOI: https://doi.org/10.1111/gean.70009
9. Baddeley A., Rubak E., Turner R. Spatial point patterns:

methodology and applications with R (Vol. 1). Boca Raton: CRC press. 2016. 828 p. DOI: https://doi.org/10.1201/b19708

10. Kondratenko Y. P., Encheva S., Sidenko I. V. Synthesis of Modular-Structured Systems for Automatization of Decision-Making Processes in transport Logistics. *Int. J. Comput.,* 2012. Vol. 11, No. 4. P. 336–344.

11. Козлов О., Кондратенко Ю., Скакодуб О. Гібридний мультиагентний метод для оптимізації нечітких комп'ютерних систем. *Управління розвитком складних систем. 2022.* Вип. 49. С. 40–51. DOI: https://doi.org/10.32347/2412-9933.2022.49.40-51

УДК 004.93

Максим Салютін
викладач кафедри
інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євген Сіденко
кандидат технічних наук., доцент,
завідувач кафедри
інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

КАСКАДНЕ СЕМАНТИЧНЕ УТОЧНЕННЯ РЕЗУЛЬТАТІВ YOLO З ВИКОРИСТАННЯМ VISION-LANGUAGE MODEL ДЛЯ РОЗПІЗНАВАННЯ ВІЙСЬКОВОЇ ТЕХНІКИ

Розвиток сучасних моделей комп'ютерного зору дозволяє виконувати автоматизоване виявлення та класифікацію об'єктів на цифрових зображеннях із високою швидкодією. Одностадійні детектори сімейства YOLO поєднують локалізацію та визначення класу об'єкта в межах одного процесу інференсу, що робить їх придатними для застосування в системах обробки аерофотознімків, матеріалів дистанційного спостереження та даних безпілотних літальних апаратів [1]. Водночас у задачах розпізнавання військової техніки ефективність таких моделей може знижуватися через значну

візуальну подібність окремих класів, зміну масштабу об’єктів, складний фон, часткові перекриття та різні ракурси спостереження. Особливістю цієї задачі є те, що правильне визначення положення об’єкта на зображенні не завжди означає правильне визначення його семантичного класу.

Одним із підходів до додаткової перевірки таких результатів є використання Vision-Language Models (VLM), здатних виконувати семантичний аналіз локалізованої області зображення відповідно до текстової інструкції [2]. Проте застосування VLM до кожної сформованої детекції суттєво збільшує обчислювальні витрати. Тому доцільним є вибіркове залучення додаткової моделі лише у випадках, коли результат первинного детектора характеризується підвищеною невизначеністю [3].

У даній роботі досліджено каскадний підхід, у якому первинне виявлення військової техніки виконується моделлю YOLO26n-640, а Vision-Language Model Qwen3-VL-4B-Instruct застосовується як додатковий засіб семантичного уточнення [4]. Основною ознакою керування подальшим маршрутом обробки є рівень впевненості первинного детектора. Запропонований механізм використовує два порогові значення: нижній поріг $\tau_L = 0{,}01$ та верхній поріг $\tau_H = 0{,}73$. Прогнози з рівнем впевненості нижче нижнього порогу відкидаються, високовпевнені результати приймаються без додаткового аналізу, а прогнози проміжного діапазону передаються до VLM. Значення порогів визначалися на окремій калібрувальній вибірці та після цього залишалися незмінними під час фінального тестування (рис. 1).

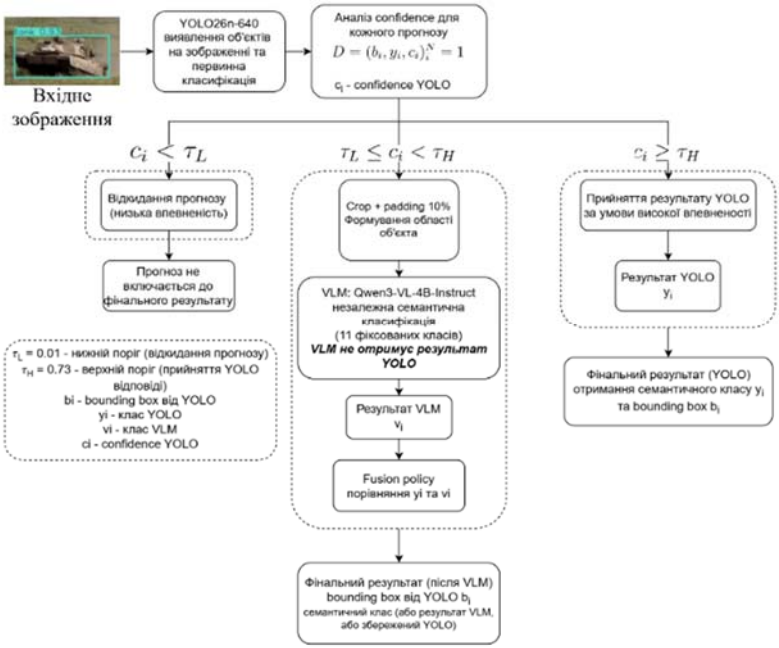


Рисунок 1. – Структурна схема каскадного семантичного уточнення результатів YOLO

Для прогнозів, направлених на додаткове уточнення, за координатами обмежувальної рамки YOLO формується окремий фрагмент зображення. Межі області збільшуються на 10% для збереження частини контекстної інформації. До Qwen3-VL-4B-Instruct передаються сформований фрагмент та перелік із 11 допустимих класів військової техніки, при цьому початковий клас, визначений YOLO, моделі не повідомляється. Таким чином, забезпечується незалежність вторинного семантичного рішення від результату первинного детектора. У запропонованій схемі VLM не виконує повторну локалізацію: координати об'єкта залишаються отриманими від YOLO, а додаткова модель використовується лише для уточнення його класу.

Для експериментального оцінювання використано набір даних військової техніки 11 класів: TANK, IFV, APC, EV, AH, TH, AAP, TA, AA, TART та SPART. Після очищення набір містив 15 978 унікальних зображень та 35 103 анотовані об'єкти. Окрему увагу приділено

запобіганню витоку даних між експериментальними вибірками. Візуально пов'язані кадри, отримані з одних послідовностей зображень, об'єднувалися у групи, після чого розподіл здійснювався на рівні груп. У результаті було сформовано незалежні навчальну, валідаційну, калібрувальну та фінальну тестову вибірки без перетину відповідних груп. Такий підхід дозволив відокремити етап визначення параметрів маршрутизації від підсумкового оцінювання методу. На фінальній тестовій вибірці первинна модель YOLO26n-640 забезпечила Precision 0,8254, Recall 0,6163, mAP@0.5 – 0,6542 та mAP@0.5-0.95 – 0,4221. Із 3385 еталонних об'єктів із прогнозами детектора було зіставлено 2545 об'єктів. Серед них 2296 мали правильно визначений клас, а 249 містили помилку класифікації, тому точність визначення класу серед зіставлених детекцій становила 90,22 %. Аналіз показав істотну різницю між рівнями впевненості правильних і помилкових результатів: середнє значення для правильних класифікацій становило 0,7383, тоді як для помилкових – 0,0640. Це стало підставою для використання confidence детектора як ознаки маршрутизації.

Параметри каскадної системи визначалися виключно на калібрувальній вибірці. До проміжного діапазону впевненості було віднесено 751 зіставлений прогноз, серед яких YOLO правильно класифікувала 423 об'єкти, а 328 прогнозів містили помилку визначення класу. Точність первинного детектора на цій підмножині становила 56,32 %. Після незалежної класифікації за допомогою VLM загальна точність становила 66,98%, при цьому Qwen3-VL-4B-Instruct правильно виправила 262 із 328 помилкових прогнозів YOLO, тобто 79,88%. Отриманий результат показав, що виділений маршрутизатором проміжний діапазон дійсно містить значну частину складних міжкласових випадків, для яких застосування додаткової семантичної моделі може бути доцільним.

Після фіксації параметрів системи її роботу оцінено на незалежній фінальній тестовій вибірці. Із 2545 зіставлених прогнозів YOLO 1681 результат перевищував верхній поріг впевненості та був прийнятий без залучення VLM. Усі прогнози цієї групи мали правильний клас. Ще 650 прогнозів було направлено на додаткове семантичне уточнення, а 214 низьковпевнених зіставлених результатів відкинуто. Окремо проаналізовано 15 849 прогнозів, які не мали відповідності еталонним об'єктам. З них 12 731 прогноз було усунуто нижнім порогом маршрутизації, що відповідає рівню відсіювання 80,33%. Найбільш показовими є результати роботи VLM із помилковими класифікаціями первинного детектора. Серед 650 прогнозів, переданих на семантичне

уточнення, YOLO правильно класифікувала 517 об’єктів, а у 133 випадках визначила неправильний клас. Qwen3-VL-4B-Instruct змогла виправити 105 із цих 133 помилок, що відповідає 78,95%. Водночас аналіз показав, що безумовна заміна всіх маршрутизованих результатів прогнозами VLM є недоцільною, оскільки вторинна модель у частині випадків змінювала початково правильні рішення. Тому остаточне рішення формувалося з використанням консервативного правила об’єднання результатів YOLO та VLM.

Після застосування маршрутизації та семантичного уточнення загальна кількість збережених прогнозів зменшилася з 18 394 до 5449, тобто приблизно на 70,4%, а кількість незіставлених прогнозів – з 15 849 до 3118, або на 80,3%. Кількість помилкових класифікацій серед зіставлених результатів скоротилася з 249 до 199, що відповідає зменшенню на 20,1%. Одночасно точність визначення класу серед збережених зіставлених прогнозів збільшилася з 90,22% до 91,46%. При цьому показник відтворення локалізації знизився з 75,18% до 68,86%, що характеризує компроміс між повнотою локалізації та семантичною надійністю сформованої множини результатів. Додатковим фактором є обчислювальна вартість вторинної моделі. Середній час опрацювання одного фрагмента Qwen3-VL-4B-Instruct становив 197,4 мс, медіанне значення – 195,1 мс, а 95-й перцентиль – 269,6 мс. На фінальній тестовій вибірці VLM використовувалася приблизно для 20% первинних прогнозів YOLO. Таким чином, каскадна організація дозволяє уникати запуску найбільш ресурсоємного компонента для більшості результатів та концентрувати додаткові обчислення на підмножині невизначених випадків.

Отримані результати підтверджують доцільність використання Vision-Language Model саме як вибіркового механізму семантичного уточнення складних міжкласових помилок первинного детектора, а не як обов’язкового другого етапу для кожної детекції. Двопорогова маршрутизація дозволяє відокремлювати високовпевнені результати, відкидати значну частину низьковпевнених прогнозів та залучати додаткову модель лише у визначеному діапазоні невизначеності. Подальший розвиток підходу доцільно пов’язати з використанням адаптивних або класозалежних порогів маршрутизації та врахуванням додаткових характеристик прогнозу для оцінювання очікуваної користі від залучення вторинної моделі.

Список використаних джерел

1. Jocher G., Qiu J., Liu M., Lyu S., Akyon F. C., Kalfaoglu M. E. Ultralytics YOLO26: Unified Real-Time End-to-End Vision Models. arXiv. 2026. arXiv:2606.03748. DOI: https://doi.org/10.48550/arXiv.2606.03748

2. Bai S., et al. Qwen3-VL Technical Report. arXiv. 2025. arXiv:2511.21631. DOI: https://doi.org/10.48550/arXiv.2511.21631

3. Huseljic D., Herde M., Hahn P., Müjde M., et al. Systematic Evaluation of Uncertainty Calibration in Pretrained Object Detectors. *International Journal of Computer Vision.* 2025. Vol. 133. P. 1033–1047. DOI: https://doi.org/10.1007/s11263-024-02219-z

4. Cattelan L. F., Silva D. How to Fix a Broken Confidence Estimator: Evaluating Post-hoc Methods for Selective Classification with Deep Neural Networks. *Proceedings of Machine Learning Research.* 2024. Vol. 244. P. 547–584.

УДК 004.9

Микита Смоленський
аспірант кафедри
інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
Юрій Кондратенко
доктор технічних наук, професор,
професор кафедри
інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Євген Сіденко
кандидат технічних наук., доцент,
завідувач кафедри
інтелектуальних інформаційних систем
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ІДЕМПОТЕНТНА ОБРОБКА RFID-ПОДІЙ У МОБІЛЬНІЙ МЕДИЧНІЙ ІНФОРМАЦІЙНІЙ СИСТЕМІ

У роботі досліджено обробку повторних запитів, які виникають після RFID-ідентифікації пацієнта та втрати відповіді сервера. Запропоновано модель, у якій підтверджена користувачем операція отримує сталий ідентифікатор, а її результат і запис про обробку зберігаються в одній транзакції. У програмному експерименті порівняно запис без контролю повторів, фільтрацію за UID мітки та обробку за ідентифікатором операції. Для 1000 операцій і 500 додаткових запитів отримано відповідно 1500, 100 і 1000 записів. У досліджених синтетичних потоках останній варіант не створював дублів і не втрачав окремих операцій. Результати характеризують програмну модель і не є оцінкою фізичного RFID-зчитування або клінічної ефективності.

Ключові слова: *RFID, ідемпотентність, повторна доставка, медична інформаційна система, транзакція, узгодженість даних*

Мобільний RFID-модуль забезпечує перехід від ідентифікатора пацієнта до його цифрового запису. Архітектури поєднання UHF RFID,

NFC та медичних інформаційних сервісів розглянуто, зокрема, у роботі [1]. Після ідентифікації користувач може підтвердити відвідування або іншу облікову дію. Якщо сервер зберіг результат, але клієнт не отримав відповіді, повторна передача такого запиту здатна створити зайвий запис. Саме цей сценарій, а не читання картки пацієнта, становить предмет дослідження.

Метою даної роботи є розроблення та експериментальна перевірка програмної моделі обробки повторних RFID-пов'язаних операцій зі збереженням окремих законних дій для одного пацієнта. Внесок полягає у визначенні межі між повторною доставкою та новою операцією, формалізації правил обробки й порівнянні трьох стратегій у відтворюваному експерименті.

Ідемпотентність означає, що повторення однакового запиту має той самий передбачуваний вплив на стан сервера, що й одноразове виконання [2]. HTTP сам по собі не надає такої гарантії довільній операції POST. Використання ідентифікатора запиту та атомарне збереження результату є відомим підходом [3]. Тут його адаптовано до події, що формується після RFID-ідентифікації та підтвердження дії користувачем; новий загальний алгоритм ідемпотентності не заявляється.

Операцію подано кортежем E = (s, k, u, a, c), де s визначає простір ідентифікаторів установи, k – ідентифікатор логічної операції, *u* – UID мітки, *a* – тип дії, *c* – її контекст. У моделі *a* відповідає підтвердженню відвідування, а *c* містить окремий номер відвідування. Ключ K = (s, k) залишається незмінним під час повторної доставки. UID не підходить на роль такого ключа: один пацієнт може мати кілька різних відвідувань.

Клієнт має створювати k один раз після підтвердження дії та зберігати його разом із запитом до отримання остаточного результату. Повторне фізичне зчитування мітки саме по собі не означає нової облікової операції. Водночас нове підтверджене відвідування отримує інший *k,* навіть якщо UID не змінився. Якщо клієнт помилково генерує новий ключ при кожній повторній відправці, серверний контроль не може встановити тотожність наміру.

Сервер порівнює канонічне представлення полів операції зі збереженим запитом. Для нового K він створює обліковий запис і квитанцію обробки в одній транзакції. Для відомого K з тим самим вмістом повертає попередній результат. Повторне використання K з іншим вмістом дає конфлікт без нового запису. Цільовий інваріант має вигляд $N(K) \leq 1$, де N(K) – число облікових ефектів для ключа K. Окремо перевіряється повнота: кожна коректна логічна операція з потоку повинна мати один результат.

Для перевірки створено автономну модель на Python 3.12.14 і SQLite 3.53.1. Обліковий ефект представлено рядком таблиці, а

квитанцію – рядком окремої таблиці з унікальним ключем. Транзакція BEGIN IMMEDIATE охоплює перевірку ключа та обидва записи. SQLite надає механізм атомарної фіксації транзакцій [4]; прикладна умова тут полягає в тому, що ефект і квитанція належать тій самій базі даних.

Кожен вхідний потік містив 1000 різних операцій для 100 синтетичних UID, тобто по 10 операцій на мітку. До нього додавали 0, 100, 300 або 500 точних копій запитів, вибраних без повернення, після чого порядок перемішували. Додаткові запити становили 0, 10, 30 або 50% від початкових 1000 операцій. Це не частка повторів серед усіх доставок: для останнього випадку вона дорівнює 500/1500, або 33,3%. Використано 30 фіксованих початкових значень генератора від 0 до 29.

Порівнювали три стратегії: безумовний запис кожного запиту; збереження лише першого запиту для UID протягом усього потоку; контроль за K із перевіркою вмісту. Загалом виконано 120 потоків для кожної стратегії, або 360 запусків і по 147 000 доставок на стратегію. Основні серії використовували базу в пам’яті. Вимірювали зайві записи та втрачені логічні операції, а не час відповіді. Код, початкові значення, приклад потоку та результати кожного запуску збережено разом із матеріалами роботи.

У стратегії без контролю кожний додатковий запит утворював зайвий запис: відповідно 0, 100, 300 і 500 дублів. Контроль за K у всіх 30 порядках для кожного рівня повторів зберігав 1000 операцій без зайвих записів. Залежність наведено на рис. 1. Незмінність підсумкових кількостей між порядками очікувана для заданого набору операцій; ці серії не доводять універсальної стійкості до всіх можливих відмов.

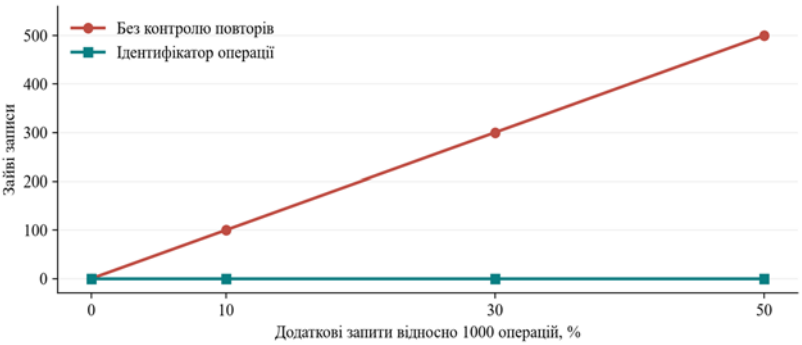


Рисунок 1. – Зайві записи за різної кількості повторних доставок

Фільтрація за UID залишала лише 100 записів і втрачала 900 окремих операцій незалежно від кількості повторів. Порівняння для 1500 доставок наведено на рис. 2. Отже, показник «нуль дублів» недостатній: його можна отримати ціною втрати коректних дій. Контроль за ідентифікатором операції задовольнив обидві перевірені умови, тоді як два базові варіанти порушували одну з них.

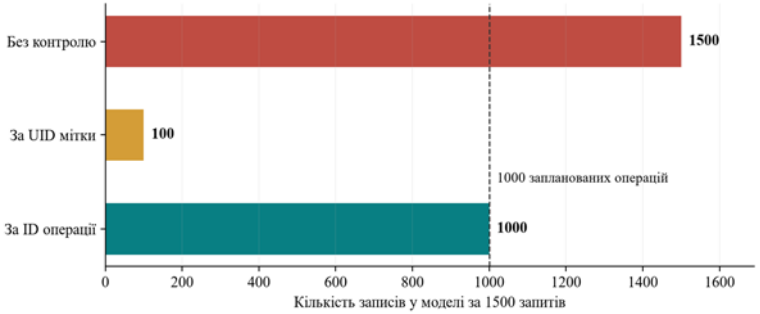


Рисунок 2. – Результати трьох стратегій для 1000 операцій і 500 повторів

Додаткові сценарії виконано по 20 разів із файловими базами даних. Контрольований виняток після створення ефекту, але до запису квитанції, спричиняв відкат; після повторного відкриття бази повтор запиту створював один результат. Якщо відповідь відкидалася після фіксації транзакції, повтор після відкриття бази повертав наявний результат без другого ефекту. Зміна контексту при незмінному K давала конфлікт у 20 із 20 випадків.

У 20 окремих перевірках два потоки через різні з’єднання одночасно подавали однаковий запит до файлової бази. Кожного разу модель створювала один ефект і повертала однаковий номер результату обом викликам. Це локальна перевірка конкуренції двох з’єднань, а не навантажувальне випробування. Контрприклад із новим ключем для того самого наміру створив два записи та підтвердив залежність гарантії від поведінки клієнта.

Межі дослідження охоплюють одну базу, довірений простір ідентифікаторів та збереження квитанцій протягом усього запуску. Виняток і втрата поверненого результату моделювали окремі відмови; фізичного обриву мережі, аварії ОС чи знеструмлення не відтворювали. Вимірювання не характеризують RFID-зчитувач, мобільний застосунок, автентифікацію або клінічні наслідки. Перед

інтеграцією потрібно визначити строк збереження ключів, правила доступу та узгодження зовнішніх ефектів, які не входять до транзакції SQLite.

Таким чином, у дослідженій моделі сталий ідентифікатор операції та спільна транзакція для ефекту і квитанції забезпечили відсутність дублів без втрати законних дій. На відміну від фільтрації за UID, підхід зберігає повторні відвідування одного пацієнта. Наступним етапом є реалізація сталої черги на мобільному клієнті та перевірка цього протоколу через реальний API з перериваннями зв'язку.

Список використаних джерел

1. Bianco G. M., et al. (2023). UHF RFID and NFC Point-of-Care – Architecture, Security, and Implementation. *IEEE Journal of Radio Frequency Identification.* DOI: 10.1109/JRFID.2023.3268422

2. Fielding R., Nottingham M., & Reschke J. (2022). HTTP Semantics. RFC 9110, section 9.2.2. DOI: 10.17487/RFC9110

3. Amazon Web Services. Making retries safe with idempotent APIs. The Amazon Builders' Library. https://aws.amazon.com/builders-library/making-retries-safe-with-idempotent-APIs/ (дата звернення: 19.10.2026).

4. SQLite. Atomic Commit In SQLite. https://www.sqlite.org/atomiccommit.html (дата звернення: 19.10.2026).

ПІДСЕКЦІЯ
ГЕОДЕЗІЯ, ЗЕМЛЕУСТРІЙ, КАДАСТР: ТЕНДЕНЦІЇ РОЗВИТКУ В СУЧАСНИХ УМОВАХ

УДК 332.7:339.138:005.57

Олена Лазарєва
доктор економічних наук, професор,
професор кафедри управління земельними ресурсами
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

НЕТВОРКІНГ ЯК ІНСТРУМЕНТ ВЕДЕННЯ ЗЕМЕЛЬНОГО БІЗНЕСУ

На сьогоднішній день нетворкінг є одним з ключових інструментів для пошуку інвестиційних ресурсів, партнерів для ведення бізнесу, для вирішення в цілому бізнес завдань, а також налагодження важливих зв'язків для досягнення поставлених цілей. Крім того, нетворкінг дозволяє знаходити нові бізнес-перспективи, обмінюватися досвідом та знаннями, що в цілому сприятиме ефективному вирішенню бізнес-питань.

Відповідно, метою нашого дослідження є вивчення особливостей нетворкінгу саме у сфері земельного бізнесу.

Нетворкінг за своєю сутністю спирається на «теорію шести рукостискань» [1], яку у 1969 році висунули американські вчені Джеффрі Треверс і Стенлі Мілгром, відповідно до якої будь-які дві людини на Землі пов'язані коротким ланцюгом спільних знайомих.

Нетворкінг за своєю сутністю є таким принципом спілкування в бізнес-середовищі, під час якого вдається налагодити бізнес-зв'язки та конструктивні ділові відносини, спираючись на довіру у бізнес-середовищі та специфічні локальні взаємозв'язки.

Якщо вдатись до аналізу закордонного досвіду нетворкінгу у сфері земельного бізнесу, то можна побачити, що він ґрунтується на створенні як цифрових сервісів, так і онлайн-платформ та інфраструктурних систем, що дають змогу забезпечувати ефективну взаємодію, а також обмін даними між землекористувачами, ІТ-фахівцями, землевпорядниками, бізнес-партнерами тощо.

Так, за кордоном нетворкінг в землекористуванні представлений такими спеціалізованими міжнародними мережами та платформами як:

- European Land Tenure Network [2], яка об'єднує експертів у сфері управління земельними ресурсами, сприяючи максимізації їхнього довгострокового соціального та екологічного впливу на забезпечення стійкості та землеустрій майбутнього;
- UN-GGIM – Експертний комітет ООН з глобального управління геопросторовою інформацією, який є головним міжурядовим консультативним органом Організації Об'єднаних Націй у сфері геопросторових інформаційних ресурсів, що забезпечує розробку та впровадження Комплексної структури геопросторової інформації, яка базується на концепції пошарового моделювання реального світу, проводить регулярні міжнародні конгреси;
- FIG – Міжнародна федерація геодезистів, яка просуває професійну практику, стандарти у сфері геодезії [3], організовуючи щорічні конгреси, які є найбільшими нетворкінг-платформами для фахівців із кадастру, оцінки земель та землевпорядкування.

Крім того, нетворкінг у сфері земельного бізнесу представлений міжнародними науково-практичними та грантовими консорціумами, такими як Консорціум RISE4UKRAINE [4], що являє собою міжнародне об'єднання 16 партнерів із Європи та України, створене для реалізації науково-інноваційного проєкту «Recovery and Innovation through Sustainable Ecosystems for Ukraine», яке сприяє справедливому, зеленому та орієнтованому на людей відновленню економіки України.

Важливим у сфері нетворкінгу є і проведення міжнародних конференцій та симпозіумів, які об'єднують інвесторів для подальшої розробки бізнес-сценарію чи бізнес-ідеї, колег та потенційних партнерів з усього світу в одному просторі.

Отже, як бачимо, на міжнародній арені нетворкінг дозволяє об'єднати провідних науковців, дослідників та практиків заради вивчення міжнародного досвіду, інноваційних підходів та цифрових рішень.

Нетворкінг у сфері земельного бізнесу практикується під час пошуку партнерів для бізнесу та потенційних інвесторів, постачальників, для вирішення різноманітних задач у сфері землекористування.

З цього приводу зазначимо, що Україні діють різноманітні спеціалізовані онлайн-платформи, де підприємці можуть спілкуватися та знаходити нові можливості для співпраці, такі як Vkursi Zemli

(сервіс для роботи із земельними даними), Feodal.Land (спеціалізована IT-платформа для поточного контролю та управління земельним банком), LandInvest (онлайн-платформа для проведення земельних торгів, купівлі, продажу та інвестування у сферу землекористування).

Відповідно, основними стратегічними напрямками взаємодії нетворкінгу та земельного бізнесу можуть бути:

– взаємодія із місцевими громадами та об'єднаними територіальними громадами, завдяки чому шляхом об'єднання владних ресурсів, громадського простору концентруються зусилля на забезпеченні ініціатив для вирішення нагальних питань у сфері земельного бізнесу, а також створення спільних проєктів повоєнного відновлення;

– муніципальне партнерство шляхом спільного використання ресурсів, а також міжнародне партнерство заради залучення грантових коштів, тим самим створюючи спільні проєкти, які сприятимуть веденню земельного бізнесу на засадах прибутковості;

– проведення бізнес-форумів та круглих столів шляхом проведення зустрічей як і владних структур, так і бізнес-партнерів, які націлені на просуванні бізнес-ідеї, пошуку інвестиційних ресурсів для прибутковості ведення бізнесу на землі;

– - залучення капіталу та інвестицій, під час чого фінансування стає результатом взаємовигідних стосунків у сфері бізнесу, забезпечуючи там самим гарантії прибутковості для бізнесу та інвестора;

– доступ через мережі комунікативних зв'язків (місцевих землевпорядників, нотаріусів та юристів) до даних про власників землі, які планують її продаж, ще до того, як об'єкт земельного бізнесу з'являється на ринку;

– створення земельних хакатонів заради розробки інноваційних рішень, цифрових продуктів або стратегії у сфері землекористування, зо сприятиме розробці таких IT-продуктів для управління територіями, які б були здатні розробляти проєкти просторового планування та відновлення територій, аналізувати знімки із дронів для пошуку вибухонебезпечних предметів на полях тощо, що є вкрай необхідних в умовах повноцінного військового вторгнення.

Окреслені особливості нетворкінгу у сфері земельного бізнесу за умови їх емерджентної взаємодії забезпечуватимуть отримання нових можливостей для розширення бізнесу, оновлення бізнес-стратегій, фокусуванні зусиль для місцевих ініціатив.

Список використаних джерел

1. Нетворкінг: переваги корисних знайомств для бізнесу. 08.08.2024. URL: https://online.novaposhta.education/blog/netvorking-perevagi-korisnih-znajomstv-dlya-biznesu (дата звернення 14.10.26).

2. Європейська мережа CLT. Міжнародний центр громадських земельних трастів. URL: https://www.cltweb.org/organizer/european-clt-network/ (дата звернення 14.10.26).

3. Міжнародна федерація геодезистів. (FIG). International Science Council. URL: https://uk.council.science/member/fig-federation-internationale-des-geometres/ (дата звернення 14.10.26).

4. Консорціум RRR4U. Український кліматичний офіс. URL: https://ukrainian-climate-office.org/stakeholder/konsortsium-rrr4u/ (дата звернення 14.10.26).

УДК 332:143

Ярослав Матвєєв
аспірант кафедри управління земельними ресурсами
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

СВІТОВИЙ ДОСВІД ЗБАЛАНСОВАНОГО РОЗВИТКУ ТЕРИТОРІЙ ТА МОЖЛИВОСТІ ЙОГО ВИКОРИСТАННЯ В УКРАЇНСЬКІЙ ПРАКТИЦІ

Формування конкурентоздатного економічного простору нині потребує збалансованого підходу до побудови стратегії і тактики розвитку територій. Це передусім стосується землекористування, враховуючи те, що земля є незамінним засобом виробництва, і потребує дбайливого та раціонального використання.

Тому з огляду на важливість та багатофункціональність землекористувань проаналізуємо світові орієнтири розвитку територій, обґрунтувавши його пріоритетні можливі сценарії в українській практиці. Зазначене і сприяло вибору теми дослідження, окресливши його мету, якою є розгляд світового досвіду розвитку територій та вивчення можливостей його використання на засадах збалансованого підходу.

Якщо говорити про збалансований розвиток територій, то, наприклад, у Німеччині землі в основному перебувають у приватній власності, а держава обмежує право землеволодіння для приватних осіб до 20 га [1]. В цій країні державне регулювання землекористування відбувається шляхом надання субсидій та консультацій з питань ведення господарювання на землі на засадах збалансованості дотримання сівозмін, що свідчить про дотримання принципу охорони грунтів, підтримання їх родючості. У Швеції збалансований розвиток територій грунтується на засадах автономії, при якій місцеві громади самостійно приймають рішення щодо розвитку території [2, с. 24]. При цьому взаємодія між місцевими органами влади та жителями громад сприяє забезпеченню ефективному розвитку територій, дотримуючись збалансованого підходу. Що стосується США, то в цій країні відсутнє єдине законодавство, яке регламентує розвиток землекористування, дотримуючись збалансованого підходу. Відповідно кожен штат в цій країні сам розробляє свої програми розвитку територій та їх раціональної організації [3, с. 233]. У Франції політика збалансованого розвитку територій передбачає стимулювання конвергенції регіонів, а також мінімізацію впливу негативних факторів зовнішнього середовища [4, с. 46]. При цьому підтримка розвитку проблемних регіонів пов'язана із забезпеченням державної системи контрактів із планування регіонального розвитку. Такий підхід сприяє комплексності та збалансованості території, поєднуючи державу, органи місцевого самоврядування та бізнесові структури, що забезпечуватиме більш розвиток регіонів на засадах конвергенції. В Нідерландах політика розвитку територій передбачає інтегрований підхід до планування, який передбачає дотримання контролю за використанням земельних ресурсів, а також впровадження інноваційно орієнтованих екологічних рішень [5]. Крім того, в цій країні економічні інтереси тісно пов'язані з екологічними потребами та розвитком агросектору. Що стосується стійкості до змін клімату, що притаманне даній країні, то розвиток територій передбачає розбудову дамб та інженерних споруд для захисту територій. В Іспанії політика регіонального розвитку ґрунтується на засадах рівномірного розвитку регіонів [6], забезпечуючи при цьому збалансований розвиток територій. В таких країнах Центральної та Східної Європи як Естонія, збалансований розвиток провадиться з урахуванням регіональних та загальнонаціональних інтересів. В Словенії прийнятий Закон «Про сприяння збалансованому регіональному розвитку» [7, с. 135], який регламентує привабливість територій для споживання, праці та

здійснення інвестицій. В Чехії збалансований розвиток територій передбачає зменшення відмінностей у розвитку регіонів та покращення економічної і соціальної інфраструктури [8, с. 36].

Збалансований розвиток національних, регіональних і місцевих стратегій у країнах ЄС на сьогодні грунтується документом «Territorial Agenda 2030: A future for all places» [9], передбачає виокремлення орієнтирів для стратегічного просторового планування на всіх рівнях управління територіями, таких як кращий збалансований територіальний розвиток різноманітностей Європи, наявність функціональних регіонів, інтеграцію поза кордони, циркулярна економіка та стійкі зв'язки. Не менш важливим документом, що передбачає наближення Європи до сталого розвитку є Зелена угода (European Green Deal – EGD) [10], яка спрямована на боротьбу зі скороченням викиду парникових газів і впливу зміни клімату, а також з іншими екологічними проблемами. У 2020 року Європейською комісією прийнято новий План дій щодо економіки замкнутого циклу (CEAP), який є одним із основних будівельних блоків Європейської зеленої угоди, нового порядку денного Європи щодо сталого зростання [11].

Отже, як бачимо, збалансований розвиток територій країн світового простору передбачає інтегрований підхід, що грунтується на поєднанні як економічного, так і соціального та екологічного підходів, забезпечуючи при цьому гармонійну їх взаємодії, що в цілому сприятиме збалансованості землекористування.

Що стосується практики українських реалій, то зазначимо, що проводяться заходи щодо визначення пріоритетів збалансованого розвитку в територіальних громадах, заохочується співробітництво між громадами в інтересах сталого розвитку. Так, на онлайн-конференції «Збалансований розвиток адміністративних територій: виклики та пріоритети» [12], що відбувалась в Хмельницькій області за підтримки Представництва Фонду Ганса Зайделя в Україні, наголошувалось на використанні кращих практик роботи територіальних громад області задля прагнення об'єднати зусилля заради збалансованого розвитку своїх територій.

Нині триває реалізація проєкту «Збалансований територіальний розвиток в ЄС» [13], що реалізується Донецьким національним університетом імені Василя Стуса за підтримки програми JEAN MONNET ACTIONS ERASMUS, в рамках якого висвітлюється європейський досвід формування політики розвитку територій та напрями потенційного впровадження європейських практик у розвиток територій в Україні.

Відповідно, можливими сценаріями збалансованого розвитку територій в практиці українських реалій у післявоєнний період можуть бути такі орієнтири як використання конкурентних переваг певної території, враховуючи її місцерозташування, позиціювання на ринку тощо, залучення інвестицій у повоєнних період для формування сприятливого підприємницького середовища, збалансований в цілому розвиток АПК, передбачення підвищення доданої вартості продукції місцевих виробників, що сприятиме зростанню їх прибутковості, місцева економічна кооперація мешканців громад для підтримки місцевих ініціатив, орієнтованих на збалансований підхід, підвищення активності та згуртованості мешканців громад, забезпечуючи при цьому фокусування зусиль на конкурентоздатності виробництва на землі, прийняття економічно обґрунтованих рішень щодо реалізації планів місцевого розвитку громад з урахуванням обґрунтованих показників їх динаміки, генерація знань та людського капіталу для застосування інноваційних технологій та інновацій у виробництві на землі, розробка сценаріїв економічного розвитку громади шляхом узгодження параметрів щодо доступності ресурсів та прибутковості ведення бізнесу на землі.

Врахування запропонованих сценарії збалансованого розвитку територій створить передумови для розвитку конкурентоздатного середовища на землі на засадах інноваційно-сприятливого підходу.

Список використаних джерел

1. Попрозман Н. В., Коробська А. О. Особливості міжнародного досвіду регулювання ринку земель сільськогосподарського призначення та їх адаптація до українських реалій. *Ефективна економіка*. 2020. № 11. URL: http://www.economy.nayka.com.ua/pdf/11_2020/13.pdf (дата звернення: 11.10.26).

2. Кіцак М. М., Білик Р. С. Зарубіжний досвід управління конкурентними перевагами територій. *Регіональна економіка.* № 4. 2022. С. 21–24. URL: http://re.gov.ua/re202204/re202204_021_KitsakMM,BilykRS.pdf (дата звернення: 11.10.26).

3. Хаустова В. Є., Трушкіна Н. В. Забезпечення сталого та раціонального землекористування: міжнародний досвід та українські реалії. *Бізнес Інформ. 2024.* № 1. С. 218–234. URL: https://www.business-inform.net/export_pdf/business-inform-2024-1_0-pages-218_234.pdf (дата звернення: 11.10.26).

4. Савчин І. З., Перепелюкова О. В. Міжнародний досвід регулювання конвергенції регіонального розвитку. *Бізнес Інформ*. 2024. №5. С. 43–49.

URL: https://mail.business-inform.net/export_pdf/business-inform-2024-5_0-pages-43_49.pdf (дата звернення: 11.10.26).

5. Ткаченко А. Аналіз інвестиційного клімату Нідерландів. URL: https://duan.edu.ua/wp-content/uploads/2024/03/tkachenko.pdf (дата звернення: 11.10.26).

6. Бабаєв В. М. Сучасні політики країн Європи: досвід для формування регіональної політики в Україні. *Державне будівництво. Світовий досвід державного управління: теорія та практика.* 2007. № 1. С. 1–28.

7. Куйбіда В., Ткачук А., Забуковець-Ковачич Т. Регіональна політика: правове регулювання. Світовий та український досвід ; за заг. ред. Р. Ткачука. Київ : Леста, 2010. 224 с. URL: https://www.csi.org.ua/wp-content/uploads/2010/03/reg_polit-prn-1.pdf

8. Кулик Д. Є. Європейські підходи до розробки та втілення проєктів розвитку регіонів та територій. *Contemporary transformations of social development: new challenges and perspectives.* P. 22–48. URL: https://www.reicst.com.ua/asp/article/download/monograph-06-2024-02-02/2024-02-02/1280 (дата звернення: 11.10.26).

9. European Commission. 2020. Territorial Agenda 2030 – A future for all places. Informal meeting of Ministers responsible for Spatial Planning and Territorial Development and/or Territorial Cohesion. 1 December 2020, Germany. URL: https://ec.europa.eu/regional_policy/sources/brochure/territorial_agenda_2030_en.pdf (дата звернення: 11.10.26).

10. European Commission (2019), The European Green Deal, COM (2019). URL: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52019DC0640 (дата звернення: 11.10.26).

11. A new Circular Economy Action Plan For a cleaner and more competitive Europe. Brussels, 11.3.2020 COM(2020) 98 final. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex:52020DC0098 (дата звернення: 11.10.26).

12. Збалансований розвиток адміністративних територій: виклики та пріоритети. URL: https://km-oblrada.gov.ua/zbalansovaniy-rozvitok-administrati/ (дата звернення: 11.10.26)

13. Активно триває реалізація проєкту «Збалансований територіальний розвиток в ЄС» (101127100 – BOON). URL: https://ec.donnu.edu.ua/aktyvno-tryvaye-realizacziya-proyektu-zbalansovanyj-terytorialnyj-rozvytok-v-yes-101127100-boon/ (дата звернення: 11.10.26).

УДК 528.94:528.8:004.9:622.691

Антон Мельник
кандидат географічних наук, доцент,
доцент кафедри геодезії, картографії
та управління територіями
Чернівецький національний університет
імені Юрія Федьковича
м. Чернівці, Україна
Андрій Мась
старший викладач кафедри
управління земельними ресурсами
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Марина Ячнюк
кандидат географічних наук, доцент,
доцент кафедри економічної географії
та екологічного менеджменту
Чернівецький національний університет
імені Юрія Федьковича
м. Чернівці, Україна

ІНТЕГРАЦІЯ GNSS, ДЗЗ ТА ВІДКРИТИХ ГІС У ПРАКТИЦІ ГЕОДЕЗИЧНОГО ОБЛІКУ ОБ'ЄКТІВ КРИТИЧНОЇ ІНФРАСТРУКТУРИ

Сучасний стан функціонування критичної інженерно-транспортної інфраструктури ставить високі вимоги до точності, актуальності та швидкості отримання просторових даних. Газорозподільні системи є складними просторово-розподіленими об'єктами, що охоплюють підземні та надземні трубопроводи, запірну арматуру, газорегуляторні пункти та засоби електрохімічного захисту. Традиційні підходи до обліку та геодезичного супроводу експлуатації таких мереж, що базуються на паперових планшетах або статичних кадастрових планах, не здатні забезпечити оперативне вирішення інженерних завдань і запобігання надзвичайним ситуаціям.

У зв'язку зі структурною реорганізацією газорозподільних підприємств України та створенням нових операторів мереж постає гостра потреба у відмові від коштовного пропрієтарного програмного забезпечення (зокрема MapInfo Pro) та переході на гнучкі інструменти з відкритим кодом (Open Source). Додатковим ускладнюючим

фактором під час проведення інженерно-геодезичних вишукувань є специфіка польових робіт: значна частина комунікацій проходить через міжселищні території, лісові масиви та складні елементи рельєфу, де повністю відсутній або є нестабільним мобільний зв'язок. Використання виключно хмарних або онлайн-орієнтованих ГІС у таких умовах призводить до втрати контролю над геодезичними даними, гальмує роботу польових бригад та підвищує ризики під час локалізації аварійних ситуацій.

Ключовим моментом для дослідження стали структурні трансформації в Чернівецькій області по даній тематиці. З 1 березня 2024 року функції розподілу газу перейшли до Чернівецької філії ТОВ «Газорозподільні мережі України». Ця націоналізація та перетворення колишнього АТ «Чернівцігаз» на підрозділ державного оператора спричинили перегляд технічної політики підприємства, зокрема – відмову від пропрієтарного програмного забезпечення MapInfo Pro на користь відкритого з ключем та гнучкого QGIS. Саме цей технологічний зсув став каталізатором для оптимізації застосування ГІС.

Мета даного дослідження полягала у розробці, удосконаленні та практичній апробації автономного методичного комплексу збору, векторизації та просторового аналізу геодезичних даних об’єктів газорозподільної інфраструктури на базі програмного середовища QGIS та мобільних ГІС-додатків у режимі відсутності інтернет-покриття. Територією дослідження обрано село Білоусівка, що адміністративно належить до Сокирянської міської громади Дністровського району Чернівецької області.

Для досягнення поставленої мети передбачалось розв’язання таких завдань: проаналізувати принцип роботи газової інфраструктури та специфіки виконання топографічних робіт на цих об’єктах; провести дешифрування й аналіз даних засобами OSM, JOSM, MapInfo та QGIS для визначення найефективнішої системи у газовій сфері; обґрунтувати вибір оптимального технологічного рішення щодо мобільного картографування та польової синхронізації даних.

Інженерно-геодезичний ланцюжок опрацювання просторової інформації охоплює три послідовні етапи:

1. Формування картографічно-геодезичної основи та обробка даних ДЗЗ. Первинним завданням виступає створення точної просторової підкладки. При роботі з матеріалами дистанційного зондування Землі критичним є усунення геометричних спотворень та систематичних зсувів растрового зображення відносно державної геодезичної мережі. Проведено порівняльне дешифрування та взаємне

координування матеріалів високої роздільної здатності з сервісів Mapbox Satellite та Bing Aerial Imagery. Перший показав найвищу деталізацію при дешифруванні контурів забудови та елементів проїжджої частини, тоді як знімки Bing виявилися ефективнішими для векторизації меж лісових масивів та природного рельєфу.

Векторизація просторових об'єктів здійснювалася у середовищі Java OpenStreetMap Editor (JOSM) із застосуванням спеціалізованого модуля buildings_tools, що дозволило суттєво підвищити геометричну точність побудови контурів споруд та оптимізувати топологічну коректність векторного шару. Усі шляхи сполучення було класифіковано за 7 категоріями (Unclassified, Secondary, Residential, Service, Tertiary, Track, Path), що важливо для розрахунку транспортної доступності інженерних вузлів та побудови оптимальних маршрутів аварійно-ремонтних бригад.

2. Автономне польове знімання та GNSS-координування. Для вирішення проблеми відсутності Інтернет-зв’язку виконано порівняльний аналіз мобільних продуктів QField та Mergin Maps. Дослідження довело, що для цілей інженерної геодезії та інвентаризації мереж найбільш надійним рішенням є екосистема Mergin Maps.

Основні переваги застосування вказаного комплексу в польових умовах полягають у наступному:

– безшовна синхронізація: використання уніфікованого формату GeoPackage забезпечує стабільну передачу геометрії, атрибутів та прив’язаних фотофіксацій через хмарне середовище Mergin Cloud без ризику виникнення конфліктів версій та втрати інформації;

– висока точність позиціонування: інтеграція мобільного додатку з портативними супутниковими GNSS-приймачами дозволяє фіксувати просторові координати запірної арматури, колодязів, підземних трас та газорегуляторних пунктів безпосередньо в полі з мінімальною середньоквадратичною похибкою;

– повноцінний автономний режим (offline): Інженер має змогу переглядати схеми газопроводів, вимірювати відстані та площі, створювати нові точкові й лінійні об’єкти та заповнювати паспорти обладнання без наявності мобільної мережі. Автоматична передача даних на центральний сервер відбувається при підключенні до стаціонарної мережі Wi-Fi.

3. Публікація та оперативно-переглядовий доступ. З метою надання доступу до просторових даних диспетчерським та адміністративним службам без ризику випадкового пошкодження або некоректного редагування бази геоданих задіяно модуль qgis2web.

Модуль конвертує налаштований геоінформаційний проєкт QGIS у локальну веб-сторінку на основі бібліотек Leaflet/OpenLayers (з підтримкою HTML/CSS/JavaScript). Це дозволяє відкривати інтерактивну карту із заповненими атрибутивними картками об'єктів у будь-якому веб-браузері з локального жорсткого диска без необхідності розгортання складних та вартісних серверних ГІС.

Мобільне програмне забезпечення QField з відкритим вихідним кодом функціонує як портативне розширення десктопної платформи QGIS, забезпечуючи перенесення функціоналу геоінформаційного середовища безпосередньо у польові умови. Застосунок сумісний із операційними системами Android та iOS і виступає повноцінним інструментом для координування, редагування просторової геометрії та ведення атрибутивних баз даних на об'єкті (рис. 1). Ключовою перевагою платформи є її висока автономність, що дозволяє виконувати картографічні та інвентаризаційні роботи на ділянках із відсутнім або нестабільним покриттям стільникового зв'язку (зокрема під час знімання міжселищних газопроводів у лісових масивах чи віддалених населених пунктах). Передавання геоінформаційних проєктів на мобільні пристрої оптимізовано за допомогою хмарного сервісу QField Cloud або шляхом локальної експортувальної архівації (ZIP) через спеціалізований модуль розширення QField Sync.

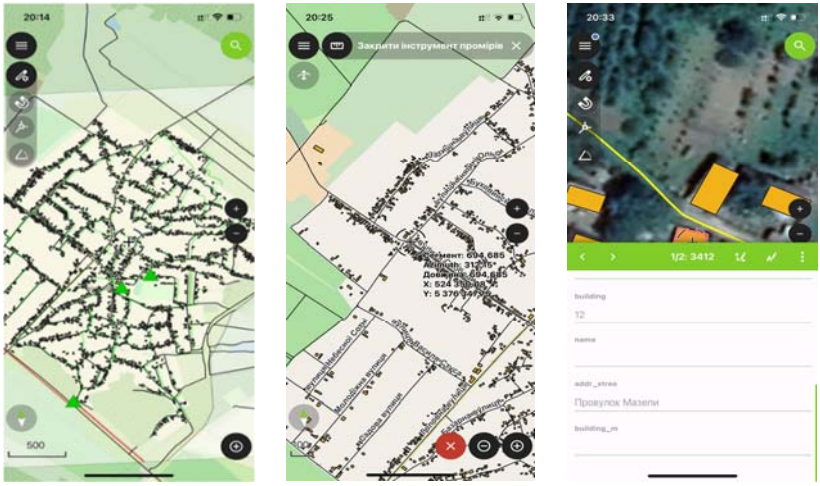


Рисунок 1. – Відображення синхронізованого геоінформаційного проекту на мобільному пристрої

Впровадження інтегрованої геоінформаційної технології «QGIS – Mergin Maps – qgis2web» забезпечує повний цикл інженерно-геодезичного супроводу об'єктів газорозподільної інфраструктури. Запропонований підхід повністю розв'язує проблему польової актуалізації геоданих в умовах відсутності покриття мережі Інтернет, гарантує високу точність супутникового позиціонування та дозволяє суттєво скоротити операційні витрати підприємства за рахунок переходу на відкрите програмне забезпечення.

Розроблені методичні рішення рекомендовані для практичного застосування у виробничих процесах Чернівецької філії ТОВ «Газорозподільні мережі України» та інших профільних організацій.

Список використаних джерел

1. Мельник А. А., Дарчук К. В., Сабадаш В. І. ГІС-технології в інженерно-геодезичному проєктуванні : навч. посіб. Чернівці : ЧНУ ім. Ю. Федьковича, 2025. 152 с.

2. Мельник А. А., Дарчук К. В., Сзкірка Л. П. Використання ГІС-технологій для аналізу геодезичного забезпечення території Дністровського району Чернівецької області. *Науковий вісник Чернівецького університету. Серія : Географія.* 2023. Вип. 842. С. 57–65. URL: http://geochnu.top/index.php/journal/article/view/142/122 (дата звернення 22.10.2026).

3. Карпінськтй Ю. О., Лященко А. А., Лазоренко-Гевель Н. Ю., Кінь Д. О., Медвецька Т. В. Методичні рекомендації, щодо оприлюднення геопросторових даних та метаданих на національному геопорталі органами місцевого самоврядування / Державнаслужба України з питань геодезії, картографії та кадастру. Київ, 2021. 49 с.

УДК 528.4

Лев Перович
доктор технічних наук, професор,
завідувач кафедри управління земельними ресурсами
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ТОЧНІСТЬ ВИЗНАЧЕННЯ ВЕРТИКАЛЬНОГО ПЕРЕМІЩЕННЯ ГРУНТОВИХ ВОД

Проблеми збереження і відтворення якості питної води є однією з найважливіших проблем людства нашої планети. В цьому аспекті особливо велика увага приділяється у Європейському Союзі, де сформульовано ряд офіційних документів щодо використання та захисту від забруднення водних ресурсів. Зокрема, Директива [2] встановлює рамки для дій Співтовариства у сфера водної політики, а Директиви [3,5] акцентують увагу на захисті ґрунтових вод від різних видів забруднення та погіршення їх стану. Рекомендації ЄС [1] та оновлена Директива [4]вказують на важливість проблеми охорони та збереження ґрунтових вод, як ключового ресурсу життєдіяльності.

Розглянемо модель водного пласта ізольованого водотривкими шарами, якиймає постійну товщину й необмежений в горизонтальному напрямку і природний потік підземних вод у відсутній. Відповідно до даної моделі час вертикального проникнення забруднених вод з поверхні землі через зону аерації товщиною m з коефіцієнтом фільтрації K і ефективною пористістю n розраховують за формулами [6]:

$$t \approx \frac{m \cdot n}{\sqrt[3]{W^2 \cdot K}} \qquad (1)$$

при незначних величинах інфільтрації ($W<K$), і

$$t \approx \frac{m \cdot n}{K} \qquad (2)$$

при просочуванні з повним насиченням пір (інтенсивна інфільтрація, витоки та т.н. ($W > K$).

При проектуванні зон санітарної охорони для міжпластових водоносних горизонтів враховують час міграції через шари слабопроникних порід, що розділяють, наприклад, забруднені ґрунтові води й міжпластовий водоносний горизонт, що експлуатується:

$$t = \frac{n \cdot m^2}{K \cdot \Delta H}, \tag{3}$$

де ΔH – різниця рівня ґрунтових вод і п'єзометричного рівня напірних вод в абсолютних позначках, м.

Виведемо математичні залежності визначення середньої квадратичної похибки санітарно-захисної зони ізольованого водного пласта.

Відомо, що для функції

$$z = f(x, y, \ldots, v) \tag{5}$$

багатьох змінних x , y, …, v її середня квадратична похибка визначається за формулою

$$\mu^2_z = (\frac{\partial z}{\partial x})^2 \mu^2_x + (\frac{\partial z}{\partial y})^2 \mu^2_y + \ldots + (\frac{\partial z}{\partial v})^2 \mu^2_v, \tag{6}$$

де $\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}, \ldots, \frac{\partial z}{\partial v}$ – частинні похідні від функції z по змінним,

$\mu_x, \mu_{y,\ldots,} \mu_v$ – середні квадратичні похибки аргументів змінних величин.

Для функції (1) маємо

$$\mu_t = \frac{t}{2} \sqrt{\frac{1}{W^2} \mu_W^2 + \frac{1}{K^2} \mu_K^2 + \frac{4}{n^2} \mu_n^2 + \frac{4}{m^2} \mu_m^2}, \tag{7}$$

для функції (2)

$$\mu_t = t \sqrt{\frac{1}{K^2} \mu_K^2 + \frac{1}{n^2} \mu_n^2 + \frac{1}{m^2} \mu_m^2}, \tag{8}$$

для функції (3)

$$\mu_t = t\sqrt{\frac{1}{\Delta H^2}\mu_{\Delta H}^2 + \frac{1}{K^2}\mu_K^2 + \frac{1}{n^2}\mu_n^2 + \frac{4}{m^2}\mu_m^2}\,. \quad (9)$$

Таким чином, отримані математичні залежності (7 – 9) дозволяють встановити час вертикального переміщення водних мас, що матиме відповідне значення при розрахунку розмірів водоохоронних зон.

Список використаних джерел

1. EEA (2022) Europe's groundwater: a key resource under pressure. Publica tions Office of the European Union. DOI: https://doi.org/10.2800/50592
2. European Parliament and European Council (2000) Directive 2000/60/EC of the European Parliament and of the Council of 23 October 2000 estab lishing a framework for Community action in the field of water policy.
3. European Parliament and European Council (2006) Directive 2006/118/EC of the European Parliament and of the Council of 12 December 2006 on the protection of groundwater against pollutionand deterioration.
4. European Parliament and European Council (2020) Directive (EU) 2020/2184 of the European Parliament and of the Council of 16 Decem ber 2020 on the quality of water intended for human consumption.
5. European Council (1991) Directive 91/676/EEC of 12 December 1991 concerning the protection of waters against pollution caused by nitrates from agricultural sources.
6. Розрахунок часу вертикального проникнення забруднених вод. URL: https://studfile.net/preview/7256779/page:2/ (дата звернення: 11.10.2026).

УДК 528.4

Анастасія Руда
старший викладач кафедри
управління земельними ресурсами
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ОГЛЯД ПІДХОДІВ ДО ЗАХИСТУ ТА ПРОСТОРОВОГО ЗОНУВАННЯ МОРСЬКОГО УЗБЕРЕЖЖЯ

Морські узбережжя України зазнають системної деградації, викликаної комбінацією абразії, зсувів та порушенням динаміки наносів, що посилюється через господарську діяльність на забудованих територіях. Застосування жорстких інженерних конструкцій є традиційним інструментом захисту берегів в Україні, але такий підхід забезпечує локальний результат та призводить до ерозії сусідніх ділянок і спричинює дисбаланс наносів. Недооцінка ландшафтної структури, екологічних функцій і рекреаційного потенціалу приморських територій при прийнятті інженерних рішень призводить до незбалансованих та функціонально обмежених прибережних середовищ [8].

Окрім населених пунктів та рекреаційних зон, значну частину приморських територій займають землі сільськогосподарського призначення. Функціонування екосистем, біорізноманіття та ефективне рослинництво значною мірою залежать від якісного стану ґрунтів. Відповідно, погіршення якості ґрунту через посушливий клімат, затоплення, дефіцит поживних речовин, а також наявність таких забруднювачів, як важкі метали, мікропластик, пестициди, гербіциди тощо, призводить до зниження продуктивності аграрного сектору [3].

Таким чином, морське узбережжя зазнає прямого впливу гідрологічних чинників, абразії, деградації ґрунтового покриву, що зумовлює необхідність диференційованого підходу до його захисту.

У міжнародному досвіді пошуку рішень боротьби з цими явищами використовуються основні підходи пристосування, захисту та відступу. Прикладом є Плани управління береговою лінією (Shoreline Management Plans, SMPs) у Великій Британії, стратегії прибережного менеджменту в Нідерландах та укази щодо прибережної ерозії (coastal erosion decrees) у США. Спільною метою таких заходів є досягнення балансу між захистом узбережжя, його розвитком та екологічною

стійкістю. Інфраструктура для захисту берегів розподіляється на три основні категорії:

1) Статична жорстка інженерія (морські дамби, хвилерізи та пірси);

2) М'яка інженерія (днопоглиблення, живлення пляжів, відновлення дюн та водно-болотних угідь);

3) Кероване відступлення (встановлення буферних або захисних зон – coastal setbacks) [1].

До 1950-х років у більшості країн світу переважали методи статичної жорсткої інженерії, однак їх висока вартість та екологічна шкода спричинила перехід до м'якої інженерії, починаючи з 1960-х років. На сучасному етапі розвитку використовується гібридна інженерія для забезпечення та захисту екологічних переваг шляхом поєднання технічних споруд з природно-орієнтованими рішеннями (NbS). Дієвість заходів захисту узбережжя залежить від різних чинників, зокрема від географічного розташування, рівня технологічного прогресу, фінансування та ефективності державного управління. Зважаючи на відмінність цих факторів у різних країнах світу реалізація заходів зі зміцнення берегової лінії має суттєві варіації [1].

В Україні існуючі заходи захисту приморських територій мають локальний характер та не забезпечують довготривалу стабілізацію узбережжя [8]. Реалізація заходів із захисту берега, проведення моніторингових досліджень та інженерно-геологічних вишукувань значно ускладняються і у більшості випадків є неможливими через обмеження доступу до прибережної смуги та акваторії моря в умовах воєнного стану. Окупація територій, мінування прибережних територій, загроза обстрілів унеможливлюють як оперативне реагування на абразійні процеси, так і реалізацію повноцінного комплексу заходів із захисту узбережжя. Наслідком цього є накопичення руйнівних проблем і неконтрольована деградація прибережних екосистем.

Чорноморському узбережжю Одеської області притаманні активні зсувні процеси через наявність крутих схилів, слабких ґрунтових шарів та підземних вод. Вагома частка комплексної системи споруд для захисту берега міста Одеси, що включає терасування, дренажні системи й хвилезахист знаходиться в аварійному стані, що викликає дефіцит наносів, розмив пляжів та посилення ерозії на суміжних ділянках. За межами міста приморські території системно деградують через відсутність цілісної стратегії та хаотичне освоєння схилів [8].

Правовий механізм захисту водних і земельних ресурсів України регламентується нормами Земельного та Водного кодексів, а також низці спеціальних законів та підзаконних актів, спрямованих на мінімізацію негативного антропогенного впливу [4]. Стаття 58 Земельного кодексу та стаття 4 Водного кодексу України визначають, що до складу земель водного фонду відносяться: території безпосередньо покриті водою; берегові смуги водних шляхів; штучно створені земельні ділянки в межах акваторій морських портів; землі зайняті прибережними захисними смугами вздовж морів, річок та навколо водойм, крім земель, зайнятих лісами; території відведені під гідротехнічні, інші водогосподарські споруди та канали, а також землі, виділені під смуги відведення для них [4, 5, 6].

Для охорони морів та внутрішніх водойм на прилеглих землях встановлюється правовий режим водоохоронних зон, прибережних захисних смуг, смуг відведення, берегових смуг водних шляхів та зон санітарної охорони. Прибережна захисна смуга визначається як частина водоохоронної зони уздовж водних об'єктів із суворішим режимом обмеження господарської діяльності [4].

Відповідно до чинного законодавства, зокрема статей 60 Земельного кодексу та 88 Водного кодексу, уздовж морів, заток і лиманів передбачено встановлення прибережної захисної смуги шириною щонайменше два кілометри від урізу води. Розмежування цих смуг та ділянок пляжної зони закріплюються у містобудівній та землевпорядній документації й вносяться до Державного земельного кадастру як обмеження землекористування. У межах населених пунктів ширина прибережних захисних смуг встановлюються згідно з комплексними планами просторового розвитку територій територіальних громад, генеральними планами населених пунктів. За їхньої відсутності – 100 метрів. Практична реалізація вищезазначених норм Земельного та Водного кодексів України ускладняється через відсутність детального технічного механізму для встановлення меж водоохоронних зон та прибережних захисних смуг [4, 5, 6].

Чинна методологія визначення ширини зазначених зон спирається на фіксованих геометричних параметрах і не враховує гранулометричний склад ґрунтів, геологічну структуру берега, активність зсувів та інтенсивність абразійних процесів.

Порядком визначення розмірів і меж водоохоронних зон та режиму ведення господарської діяльності в них, затвердженим постановою Кабінету Міністрів України від 8 травня 1996 р. № 486, передбачено механізм обчислення розмірів водоохоронних зон і регламентовано їхнє господарське використання. Проте цей нормативний документ не

визначає порядок винесення меж у натурі і не посилається на інші підзаконні акти. Водночас, згідно із Тимчасовими методичними вказівками по складанню кадастрових планів обмежень і обтяжень щодо використання земель, прибережні захисні смуги визнаються як обмеження у використанні земель [4].

Проблематику впорядкування, використання та охорони прибережних територій висвітлено у працях Л. Перовича, В. Сай, якими запропоновано розрахунок визначення ширини прибережної захисної смуги від меженного рівня води. Зазначений підхід ґрунтується на визначенні ширини смуги за допомогою коефіцієнтів, що враховують гранулометричний склад ґрунту, рівень лісистості та видовий склад лісової рослинності, параметр шорсткості за відсутності лісу, а також види земельних угідь. Розрахунок проводиться за наступною формулою[7]:

$$L_{\text{ПЗС}} = L_{\text{вим}} \cdot К_{\text{л}} \cdot К_{\text{гр}} \cdot К_{\text{е}} \cdot К_{n} \cdot С_{v} \quad (1)$$

де $L_{\text{ПЗС}}$ – оптимальна ширина прибережної захисної смуги, м;

$L_{\text{вим}}$ – встановлена Водним кодексом ширина прибережної захисної смуги, м;

$К_{\text{л}}$ – коефіцієнт переходу від листяного лісу до інших його видів;

$К_{\text{гр}}$ – коефіцієнт переходу від супіщаних до інших за гранулометричним складом ґрунтів;

$К_{\text{е}}$ – коефіцієнт переходу від північної експозиції до інших експозицій;

$К_{n}$ – коефіцієнт переходу від використання прилеглої території;

$С_{v}$ – коефіцієнт варіації [7].

Як уже зазначалось, прибережні екосистеми і території зазнають значного антропогенного впливу та кліматичних змін, негативним наслідком яких є абразія берегів. Дієвим інструментом для мінімізації цих ризиків виступає оцінювання вразливості узбережжя із застосуванням Індексу прибережної вразливості (CVI)[2].

Метод CVI, запропонований Gornitz, і у подальшому модифікований Thieler and Hammar-Klose, з часом був суттєво модифікований завдяки включенню додаткових екологічних і соціально-економічних параметрів. Сучасні підходи до класифікації показників CVI виділяють три основні категорії: фактори антропогенно-екологічного впливу на узбережжя, морфолого-

геофізичні характеристики та соціально-економічні індикатори. Проте більшість досліджень продовжує зосереджуватись на перших двох групах змінних [2].

Boumpoulis та ін. запропоновано новий методичний підхід для кількісної оцінки схильності прибережних зон до абразії та введення геотехнічних змінних у розрахунок CVI. Запропонований індекс має назву «геотехнічний індекс прибережної вразливості (GCVI)» та має низку переваг, що відрізняють її від традиційних підходів. До цих переваг належать: використання даних реального часу, заміну узагальнених геологічних характеристик на точні та кількісні геотехнічні параметри, провадження інструментів нечіткої аналітичної ієрархії та методу головних компонент, оцінювання чутливості моделі для виявлення основних чинників пливу на її побудову [2].

Світовий та національний досвід свідчить про те, що традиційні методи суворої інженерії та ізольованого будівництва захисних споруд у довгостроковій перспективі часто мають обмежену ефективність. З огляду на це, доцільно поступово переходити до комбінованих або більш гнучких підходів. Існуючу практику визначення прибережних захисних смуг, що здебільшого базується на фіксованих геометричних параметрах, варто доповнити адаптивними підходами. Врахування абразійної активності берегів, геотехнічних властивостей ґрунтів та кліматичних чинників через сучасні підходи до зонування надасть змогу підвищити ефективність регулювання землекористування. Такий комплексний підхід забезпечить збалансований розвиток приморських регіонів та оптимізацію системи просторового планування.

Список використаних джерел

1. Challenges and Lessons Learned from Global Coastal Erosion Protection Strategies / D. B. Angnuureng, et al. *iScience*. 2025. P. 112055. DOI: https://doi.org/10.1016/j.isci.2025.112055 (дата звернення: 13.10.2026).

2. Presentation and analysis of the Geotechnical Coastal Vulnerability Index and validation of its application to coastal erosion problems / V. Boumpoulis, et al. *Scientific Reports*. 2025. Vol. 15, No. 1. URL: https://doi.org/10.1038/s41598-025-85594-y (дата звернення: 13.10.2026).

3. Soil salinization due to saltwater intrusion in coastal regions: The role of soil characteristics and heterogeneity / V. Sobhi Gollo, et al. *InterPore Journal*. 2024. Vol. 1, No. 1. P. ipj260424–6. URL: https://doi.org/10.69631/ipj.v1i1nr15 (дата звернення: 13.10.2026).

4. Богданов, В. В. Правовий режим прибережних захисних смуг за законодавством України. *Науковий вісник Ужгородського*

національного університету. Серія : Право. 2025. № 2(89). С. 354–359. URL: https://doi.org/10.24144/2307-3322.2025.89.2.53 (дата звернення: 13.10.2026).

5. Водний кодекс України : Закон України від 06.06.1995 № 213/95-ВР : станом на 22 лип. 2026 р. URL: https://zakon.rada.gov.ua/laws/show/213/95-вр#Text (дата звернення: 13.10.2026).

6. Земельний кодекс України : Закон України від 25.10.2001 № 2768-III : станом на 1 жовт. 2026 р. URL: https://zakon.rada.gov.ua/laws/show/2768-14#Text (дата звернення: 13.10.2026).

7. Перович Л. М., Сай В. М. Нормативно-правове та геодезичне забезпечення кадастру земель водного фонду : монографія / М-во освіти і науки України, Нац. ун-т «Львів. політехніка». Львів : Вид-во Львів. політехніки, 2017. 128 с.: іл., табл., схеми.

8. Рогачко С. І., Гусак М. П. Ландшафтно-орієнтований підхід до захисту морських берегів України. *Herald of the Odessa National Maritime University*. 2026. Вип. 80. С. 44–54. URL: https://doi.org/10.47049/2226-1893-2026-2-44-54 (дата звернення: 13.10.2026).

УДК 332.2:332.3

Олена Лазарєва
доктор економічних наук, професор,
професор кафедри управління земельними ресурсами
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна
Олена Сандольська
стейкхолдер, директор,
ТОВ «Миколаївський
земельно-кадастровий інститут»
м. Миколаїв, Україна

ОСОБЛИВОСТІ СИТУАЦІЙНОГО ПІДХОДУ У ЗЕМЛЕКОРИСТУВАННІ

У землекористуванні необхідним є створення підгрунтя для гармонійного соціально-економічного розвитку територіальної громади та кожної людини зокрема. Необхідно враховувати, що

модель господарювання, яка сьогодні має місце на селі, не забезпечує високоприбуткове ефективне використання земельних ресурсів, не дозволяє в повній мірі реалізувати наявний виробничо-ресурсний потенціал, не є конкурентоздатною в сучасних умовах економічного розвитку. До цього слід додати й інші негативи, що зумовлені викликами повноцінної війни. Так, збитки агросектору через засмічення земель становлять 1,87 трлн грн, а внаслідок забруднення ґрунтів – 23,1 млрд грн [1].

В системі управління землекористуванням за допомогою ситуаційного підходу досягається кінцевий ефект землекористування, виключаючи ситуацію негативних впливів на нього. Відтак, вважаємо за необхідне висвітлити окремі аспекти розвитку ситуаційного підходу у землекористуванні в контексті реалій сьогодення.

На практиці ситуаційний підхід передбачає використання прямого впливу на конкретну ситуацію, тобто, певного набору обставин, які впливають на землекористування в конкретний період. З урахуванням цього ситуаційний підхід дозволяє приймати рішення не відповідно до встановлених планів роботи, а в міру виявлення потенційних проблем, що потребує значного рівня децентралізації влади, й отже, забезпечує адаптивність та гнучкість організаційної структури та швидку реакцію на умови, що змінюються.

Проте в контексті вищевикладеного зазначимо, що для ситуаційного підходу характерний такий недолік, як відсутність стратегічного планування.

Враховуючи, що стратегічне планування характеризує завжди процес, що зосереджений, в першу чергу, на проблемних питаннях, він звичайно має включати в себе і детальну оцінку наявних земельних ресурсів з погляду їх подальшої експлуатації (мається на увазі і відсоток господарської освоєності, і передбачуваність чистих парів в структурі посівних площ), і орієнтацію на практичні результати запланованих в межах дослідження дій.

Планування ринкового землекористування має враховувати інноваційні орієнтири з можливістю їх впровадження при реалізації програм розвитку територій, такі, як орієнтація на виробництво та переробку екологічно чистої продукції, що є конкурентоспроможною на внутрішніх та зовнішніх ринках, орієнтація на розвиток високорентабельних та конкурентоздатних виробництв, спрямування на створення екологобезпечної продукції з використанням сучасних інноваційно-технологічних розробок, що створюють сприятливі умови в місцях, де наразі є це можливим.

Зазначені орієнтири потребують обов'язкового врахування при розробці та реалізації програм розвитку територій, що забезпечить перехід управління землекористуванням на інноваційний шлях розвитку.

В умовах обмеженості фінансово-інвестиційних джерел фінансування в умовах повноцінної війни важливе місце займає обґрунтування інноваційних стратегій розвитку землекористування. Стратегія розвитку сільськогосподарського землекористування має включати в себе використання досягнень науково-технічного прогресу та включати комплексний підхід до планування територій.

Стратегічною метою землекористувача має бути забезпечення умови родючості ґрунту, захист його від процесів ерозії і дефляції, а для того, щоб розробити стратегічний план землекористування, потрібно спочатку провести ґрунтові та інші обстеження, розробити проект використання земель, забезпечити авторський нагляд за реалізацією проекту та знайти можливості інвестиційного забезпечення. Але, в свою чергу, проєкт використання земель буде науково обґрунтованим і правильним, якщо будуть дотримані певні правила його розробки, такі як встановлення оптимального складу земельних угідь, впорядкування сіножатей та пасовищ, еколого-економічне обґрунтування системи сівозмін та передбачення заходів з охорони ґрунтів, а також забезпечення захисту довкілля від антропогенного впливу.

Оптимальним для нашої держави (з погляду на ефективність системи управління землекористуванням), є створення єдиної кадастрово-реєстраційної системи по реєстрації земельних ділянок, об'єктів нерухомості і прав на них. Це необхідно для забезпечення в наявності необхідної інформації та планово-картографічних матеріалів, передбачення фінансових витрат, часу на розробку й запровадження системи, комплектації кадрами, забезпеченням програмно-технічними засобами.

При цьому стратегія планування сільськогосподарського землекористування, орієнтована на активізацію інноваційних чинників конкурентоспроможності землекористування, спрямована на виконання триєдиного завдання:

– безпосереднє вжиття заходів регіонального рівня щодо поліпшення якісних характеристик земельно-ресурсного потенціалу, впровадження нових екологобезпечних технологій;

– заохочення до виконання робіт на землі шляхом спрямування суб'єктів господарювання до збільшення пропозиції інноваційної екологобезпечної продукції;

– заохочення попиту аграрії на інноваційні продукти, екологобезпечні технології та створення умов до впровадження інноваційну у виробничу діяльність на землі.

Отже, заплановані заходи мають освоюватися в умовах ідентифікації регіональних проектів землекористування на основі стратегічного регіонального планування та безпосередньо фінансового планування землекористування, за рахунок залучення фінансових ресурсів та прямих іноземних інвестицій, а також власних коштів землевласників та землекористувачів.

При цьому ситуаційний підхід у землекористуванні має базуватися на методологічних положеннях управління з обов'язковою конкретизацією окремих управлінських дій відповідно до ситуації, що склалася в умовах господарювання землевласника та землекористувача.

Такий підхід до управління сільськогосподарським землекористуванням сприятиме ефективному розв'язанню наявних проблем, що виникають під час використання земельних ресурсів у сільськогосподарських підприємствах.

На регіональному рівні місцевим управлінням Держгеокадастру доцільно розробити систему визначення цілей земельної політики та пріоритетних напрямів діяльності. З метою впровадження в дію законодавчих актів із регулювання земельних питань, слід визначити корегувальні заходи для забезпечення умови раціонального землекористування. Необхідно делегувати повноваження щодо впровадження в дію запропонованих інструментів на низові (місцеві) рівні державної влади. Враховуючи те, що кожен регіон, область, район, місто, село і селище мають сприяти розроблення галузевих і регіональних програм з питань регулювання земельних відносин, необхідним є також розробка нових підходів до фінансування програм. Адже за відсутності фінансування, майже всі програми та концепції залишаються невтіленими на практиці.

Реалізацію земельної політики на районному та базовому рівнях слід зосередити у відповідних виконавчих органах адміністративних районів і територіальних громад. Структура управління земельними ресурсами має передбачати формування такої структури, яка розмежовує управлінські, виконавські та контролюючі функції між суб'єктами управління на різних організаційних рівнях. Даний підхід дозволить організувати у трьох напрямах адміністрування розподілу земельних ресурсів, організацію ефективного використання і охорони земель, розмежовувати управлінські функції між суб'єктами управління на різних організаційних рівнях.

Крім того, доцільним є забезпечення принципу субсидіарності, який полягає в прийнятті рішень та наданні публічних послуг з максимальною близькістю до споживача з урахуванням можливості їх якісного надання із забезпеченням вільного доступу осіб до отримання послуг, що стосуються угод із землею, а також призначений для того, щоб максимально забезпечити вирішення на низовому рівні тих питань, які немає необхідності передавати нагору. Відповідно до цього принципу виконання владними органами функцій із вирішення суспільних завдань відбуваються на максимально наближеному до суспільства (громади) рівні.

Варто зазначити, що деяких закордонних країнах (Франція, Литва) створюються неприбуткові агентства розвитку села, що виконують такі основні функції:

– вибір місця, планування, управління, технічна і фінансова підтримка реалізації інвестиційних заходів у сільському господарстві;

– комплексне управління земельними ресурсами в сільських районах шляхом придбання та використання земель для сільського господарства та інфраструктури, екологічних та інших громадських потреб;

– оновлення сіл, розвиток сільського господарства та територіальних громад;

– -підготовка та реалізація комплексних регіонального розвитку [2, с. 19].

Для України доцільним було б створення подібних організацій, які б сприяли розвитку сільських територій. До перспектив розвитку цих агентств можна віднести забезпечення можливості для фермерів використовувати капітал на придбання споруд і техніки, консолідація аграрної та виробничої структури, уникнення непередбачуваних цінових коливань на ринку землі.

Органи державної влади та місцевого самоврядування мають здійснювати державний нагляд та контроль за раціональним землекористуванням, захищати права землевласників та землекористувачів. З метою забезпечення достовірності та правдивості слід постійно оновлювати дані про земельні ділянки, про стан землекористування на регіональному рівні. Діяльність органів виконавчої влади має передбачати проведення стратегічного аналізу, прогнозування та моніторингу земельних угідь.

Отже, підсумовуючи вищевикладене, зазначимо, що ситуаційний підхід у землекористуванні має передбачати використання прямого

впливу на конкретну ситуацію, тобто певного набору обставин, які впливають на землекористування в даний період.

Незважаючи на те, що для ситуаційного підходу характерна відсутність стратегічного планування, зазначено, що він має включати в себе детальну оцінку земельних ресурсів з точки зору їх подальшої експлуатації та орієнтацію на практичні результати запланованих в межах дослідження дій. Відтак, планування ринкового землекористування повинно враховувати такі інноваційні орієнтири з можливістю їх впровадження при реалізації програм розвитку територій як виробничі, агропромислові та технічні, що забезпечить перехід управління землекористуванням на інноваційний шлях розвитку. Ситуаційний підхід у землекористуванні має орієнтуватися на методологічних положеннях управління з конкретизацією окремих управлінських дій відповідно до ситуації, яка склалася в умовах господарювання.

Список використаних джерел

1. Збитки довкіллю України від великої війни сягнули 7,2 трлн грн. [29.07.2026]. URL: https://skilky-skilky.info/zbytky-dovkilliu-ukrainy-vid-velykoi-viyny-siahnuly-7-2-trln-hrn/ (дата звернення 15.10.2026).

2. Шафранська Л. Українські фахівці переймали досвід Євросоюзу з управління землями та планування землекористування. *Землевпорядний вісник. Науково-виробничий журнал*. № 1. 2015. С. 18–21.

УДК 528.8:631.459

Сергій Чорний
доктор сільськогосподарських наук, професор,
професор кафедри управління земельними ресурсами
Чорноморський національний університет
імені Петра Могили
м. Миколаїв, Україна

ГЕОПРОСТОРОВИЙ АНАЛІЗ ЕРОЗІЙНОЇ НЕБЕЗПЕКИ ЗЕМЕЛЬ ІЗ ВИКОРИСТАННЯМ МОДЕЛІ RUSLE ТА ГЕОІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Деградація ґрунтів є однією з найгостріших проблем, яка безпосередньо впливає на продовольчу безпеку, стале використання природних ресурсів та стан водойм. Серед деградаційних процесів особливе місце в Україні посідає водна ерозія. Водночас сучасна система управління земельними ресурсами дедалі більше орієнтується на рівень адміністративних районів і територіальних громад, які після адміністративно-територіальної реформи отримали суттєві повноваження щодо прийняття управлінських рішень.

За таких умов особливого значення набуває просторово деталізована інформація про потенційну ерозійну небезпеку, яка може бути використана під час розроблення комплексних планів просторового розвитку територій. Алгоритм дослідження повинен передбачати послідовне виконання етапів геопросторового аналізу ерозійної небезпеки земель територіальної громади із використанням адекватних математичної моделі водної ерозії та геоінформаційних технологій. Такий алгоритм нескладно реалізується в середовищі геоінформаційних систем, зокрема, QGIS та GRASS.

Емпірична математична модель RUSLE (Revised Universal Soil Loss Equation), яка була розроблена для оцінки та прогнозування середньорічних втрат ґрунту внаслідок водної ерозії має загальний наступний вигляд (Wischmeier, Smith, 1978; Benavidez, at al., 2018; Лук'янчук, 2019):

$$A = R \times K \times LS \times C \times P$$

де
A − потенційні середньорічні втрати ґрунту, т·га$^{-1}$·рік$^{-1}$;
R − фактор ерозійної сили опадів (МДж·мм·га$^{-1}$·год$^{-1}$·рік$^{-1}$);
K − фактор еродованості ґрунту (т·год·га$^{-1}$·МДж$^{-1}$·мм$^{-1}$);

LS − топографічний фактор (не має розмірності);
C − фактор рослинного покриву (не має розмірності);
P – фактор протиерозійних заходів (не має розмірності).

Просторова реалізація цієї моделі в QGIS для певної території складається з кількох етапів.

На першому етапі формується вихідна інформаційна база дослідження:

– цифрова модель рельєфу, з використанням Copernicus DEM GLO-30 із просторовою роздільною здатністю 30 м (Copernicus Data Space Ecosystem);

– інформація про ґрунтовий покрив, яка отримується з карти ґрунтів Миколаївської області масштабу 1:200 000;

– стан рослинного покриву визначається за мультиспектральними супутниковими знімками Sentinel-2 MSI;

– ерозійної сила опадів, отримуються з бази NASA POWER (https://power.larc.nasa.gov);

– просторові межі території дослідження визначаються за даними Державного земельного кадастру України.

Усі матеріали приводяться до єдиної системи координат WGS 84/UTM zone 36N (EPSG: 32636), обрізаються по межах території, яка вивчається та узгоджуються за просторовою роздільною здатністю. Підготовлені шари подальшому використовуються як основа для подальшого розрахунку факторів моделі RUSLE.

Останній етап передбачає розрахунок окремих факторів RUSLE згідно рівняння, яке приведене вище та створення окремих растрових шарів факторів R, K, LS, C і P в QGIS, які інтегруються у ГІС-середовищ за допомогою растрового калькулятора. Результатом цього етапу є фінішна растрова карта потенційних втрат ґрунту при P = 1,0 (відсутність спеціальних протиерозійних заходів), з просторовою роздільною здатністю в 30 м. Отриманий растр втрат ґрунту, класифікується за п’ятьма градаціями інтенсивності ерозійних процесів: незначна, слабка, помірна, сильна та критична (О. Світличний, А. П’яткова, 2010).

Попередні розрахунки та картування території деяких громад Миколаївщини показали, що рівнинний рельєф Причорноморської низовини приводить до того, що землі першої групи, з потенціалом незначних втрат ґрунту (менше 1,0 т/га/рік) просторово представлений переважно великими суцільними масивами сільськогосподарських угідь із відносно сприятливим поєднанням крутизни схилів, довжини

схилів, ґрунтових властивостей та рослинного покриву. Другим за площею є клас слабких втрат (1,0-5,0 т/га/рік). Осередки слабких втрат (1,0-5,0 т/га/рік) мають переважно мозаїчне просторове поширення та часто прив'язані до не дуже крутих схилових ділянок, меж сільськогосподарських полів.

Території з втратами 5-10 т/га/рік (помірна ерозія) вже слід розглядати як зони підвищеного ризику. Їх площа як правило не перевищує 10% території в регіоні. На таких ділянках необхідне детальніше обстеження рельєфу та фактичного стану ґрунтового покриву. Залежно від виробничих та природних умов ці землі можуть застосовуватися в землеробстві, із застосуванням контурного землеробства, смугового розміщення культур, залуження окремих схилових ділянок та створення штучних водорегулювальних елементів.

Ділянки з втратами 10-20 т/га/рік (сильна ерозія) потребують пріоритетного втручання землевпорядників та служб охорони Грунів. Ці ділянки мають найбільш локалізований характер і переважно формують смуги та плями в межах земель, які мають великі ухили та довгі схили. Їх наявність свідчить про підвищення потенційної інтенсивності площинного змиву ґрунту порівняно з фоновими територіями. Їх доцільно розглядати при плануванні протиерозійної організації території, як окремі робочі ділянки, де інтенсивне рільництво може бути недоцільним. За необхідності такі землі можуть переводитися під багаторічні трави, лісосмуги або інші види використання з постійним рослинним покривом.

Найбільшої уваги потребують території з потенційними втратами понад 20 т/га/рік. Саме ці ділянки доцільно визначити як території першочергового протиерозійного обстеження. Попри незначну площу (1-2% від загальної території), ці ділянки вже мають найбільш еродовані ґрунти. Вони характеризуються високим потенціалом потенційного змиву та можуть бути пов'язані з найбільш ерозійно небезпечними схилами та локальними концентраторами поверхневого стоку. Саме ці ділянки в першу чергу підлягають вилученню з активного землекористування, залісненню та залуженню. Перед прийняттям управлінських рішень результати картування водно-ерозійної моделі RUSLE необхідно перевірити ерозійну ситуацію за актуальними даними дистанційного зондування Землі та провести тур польових обстежень. За підтвердження високої ерозійної небезпеки тут можуть бути рекомендовані консервація окремих земельних ділянок, створення або відновлення лісових смуг, облаштування водовідвідних та водорегулювальних елементів.

Практична цінність створеної методики полягає також у можливості її подальшої інтеграції до геоінформаційної системи громади. У такому разі інформація про потенційні втрати ґрунту може бути поєднана з кадастровими ділянками, категоріями земель, видами угідь, різноманітними картами, пов'язаними з родючістю ґрунтів, цифровою моделлю рельєфу, фактичним землекористуванням тощо. Це дозволяє формувати для кожної земельної ділянки або масиву додаткові показники ерозійного ризику та використовувати їх під час планування просторового розвитку громади, підготовки документації із землеустрою, оптимізації структури сільськогосподарських угідь тощо.

Список використаних джерел

1. Лук'янчук К. А. Геоінформаційне моделювання розвитку ерозійних процесів в Середньоподільській височинній області (Шумський район Тернопільщини). *Часопис картографії.* 2019. Вип. 1(20). С. 50–68.

2. Світличний О. О., П'яткова А. В. Прикладне ерозієзнавство : навч. посіб. Одеса : ОНУ, 2020. 268 с.

3. Benavidez R., Jackson B., Maxwell D., Norton K. A review of the (Revised) Universal Soil Loss Equation ((R)USLE): with a view to increasing its global applicability and improving soil loss estimates. Hydrology and Earth System Sciences. 2018. Vol. 22, No. 11. P. 6059–6086.

4. Wischmeier W. H., Smith D. D. Predicting Rainfall Erosion Losses: A Guide to Conservation Planning. Washington, DC : U.S. Department of Agriculture, 1978. 58 p. (Agriculture Handbook ; No. 537).

ДЛЯ НОТАТОК

Редактор *А. Фесенко.*
Комп'ютерна верстка, дизайн обкладинки *Я. Олисько.*
Друк *С. Волинець*. Фальцювально-палітурні роботи *О. Мішалкіна.*

Підп. до друку 11.11.2026.
Формат 60x84 $^{1}/_{16}$. Папір офсет.
Гарнітура «Times New Roman». Друк ризограф.
Ум. друк. арк. 22,75. Обл.-вид. арк. 16,4.
Тираж 10 пр. Зам. № 7423.

Видавець і виготовлювач: ЧНУ ім. Петра Могили.
54003 м. Миколаїв, вул. 68 Десантників, 10.
Тел.: +380 (512) 50-03-32, +380 (512) 76-55-81,
e-mail: rector@chmnu.edu.ua
Свідоцтво суб'єкта видавничої справи ДК № 6124 від 05.04.2018